Vol.2011-GN-79 No.16
2011/3/17

ソフトウェア開発会議における ステークホルダと関心事の可視化ツール

鵜飼孝典[†1,†2] 林 晋平[†2] 佐伯元司[†2]

ソフトウェア開発は, ステークホルダによる協調作業である. プロジェクトマネージャや分析者にとって, ステークホルダの関心事を理解し, ステークホルダの偏りや不足などの潜在的な問題を知っておくことは重要である. 我々は, システムの信頼性, 効率性, 使用性等のシステム品質に関する重要な関心事をもれなく獲得するためにステークホルダの不足や偏りを発見することを支援するツールを提案している. 提案ツールでは, ゴール指向分析手法の一つである AGORA の成果物であるゴールグラフを対象に, 要求分析作業中にステークホルダと, ステークホルダのシステム品質に関する関心事の関係をアンカーマップを使って可視化する. 本論では, ステークホルダとステークホルダの関心事の関係を可視化するツールの実装と, その評価実験について, 少ないサンプルではあるが, 良好な結果が得られたことを述べる.

Visualizing Stakeholder Concerns for Software Development Meeting

Takanori Ugai,[†1,†2] Shinpei Hayashi[†2] and Motoshi Saeki[†2]

Software development is a cooperative work by stakeholders. It is important for project managers and analysts to understand stakeholder concerns and to identify potential problems such as imbalance of stakeholders or lack of stakeholders. A tool which visualizes the strength of stakeholders' interest of concern on two dimensional screens has been proposed. The proposed tool generates an anchored map from an attributed goal graph by AGORA, which is an extended version of goal-oriented analysis methods. It has information on stakeholders' interest to concerns and its degree as the attributes of goals. In this paper, an integrated implementation of an anchored map viewer and an goal graph editor is shown and experimental evaluation is described. Results show the tool's usefulness.

1. はじめに

ソフトウェア開発は, 開発しようとするソフトウェアに関わるステークホルダによる協調作業であるため, 適切なステークホルダを発見することが重要である[1]. また, ソフトウェアの開発を成功させるためには, ユーザの要求を正確に理解することが必須であるが, ソフトウェアを開発する対象領域に関する知識は, ステークホルダに依存する部分が多いため必要なステークホルダが欠けていると, 高品質な要求仕様を作成することはできなくなる. ロンドンの救急車のサービスの失敗事例[2] は, 現場のステークホルダに要求を聞かなかったために起こった典型例である. プロジェクトにおいて, コストに関心を持つステークホルダが少なく, 多くのステークホルダが使用性にばかり興味を示し, 要求を出していたとすると, 開発費が予算内に納まらなくなる. また, 長期運用予定のシステムの開発において, 保守性に十分関心を払わなければ, 予想外の運用費がかかることになる. 一方, 多くのステークホルダが関係するプロジェクトにおいて, 多くの人が信頼性に興味を持たない場合, 興味を持つステークホルダが少数でも同定できれば, その少数のみで要求獲得の作業を行なうことで, 効率的に作業を進めることができる.

このようなことから, プロジェクトマネージャや分析者が, 品質に関するステークホルダの関心事を理解し, ステークホルダの関心時に対する偏りや不足から起こりうる潜在的な問題に注意を払いながら作業を効率化することは非常に重要である. このような目的のために, 要求と, それに関わるステークホルダの対応関係を示す表が, プロジェクトマネジメントにおいて良く使われる[3] が, 先に上げたような潜在的な問題の原因となりうるようなステークホルダの偏りを知るためには, 特にステークホルダが多くなると多くの時間を必要とする. プロジェクトに対する考えられるリスクは数多いため, 一つ一つの小さなステークホルダの偏りや不足の観点を短時間でチェックできることは, 非常に重要である. Sharp らは, システムに関わる全てのステークホルダを発見する技術を提案している[1] が, 開発に入る前に行なうことを想定しており, それに基づいて, ステークホルダを集めたとしても, 実際に集められたステークホルダで, 十分であったかどうか知ることは難しい.

このような問題に対し, 我々は, ゴール指向分析手法の一つである AGORA[4),5)] を対象に, 要求分析作業中にステークホルダとステークホルダの品質に関する関心事の関係をアンカー

†1 富士通研究所/Fujitsu Laboratories Ltd.
†2 東京工業大学/Tokyo Institute of Technology

© 2011 Information Processing Society of Japan

マップ[6),7)] を使って可視化ツールの提案を行なっている[8)]. このツールにより, ステークホルダの関心時に対する偏りや関心時に対するステークホルダの不足の発見を短時間で行なうことができる. ここでは, ステークホルダの不足を発見するということは, 関心を持つステークホルダがいない関心時を同定することであり, 特定の関心時に対して関心を持つステークホルダが不足していることを意味する.

文献 8) では, 提案ツールの基本的なアイディアが記載されている. しかしながら, 提案ツールを使うことで短時間で正確にステークホルダの不足や偏りを同定できるできるということの効果は, 評価検証されていない. 本稿では, ステークホルダとステークホルダの関心事の関係を可視化するツールの評価実験について述べる. この評価実験では, 被験者実験により, 既存のステークホルダと課題の対応表よりも, 短時間で正確にステークホルダの不足や偏りを同定するために役立つことを示す.

以下, 本稿では, 2 章で, AGORA とアンカーマップ, ステークホルダの関心事の概要を述べた後, 3 章で, ゴールグラフからアンカーマップを作成する方法を述べる. さらに 4 章で実装, 5 章で提案ツールの評価実験について述べる. 最後に 6 章で他の研究との関連を述べ, 7 章でまとめと今後の課題を述べる.

2. 準　　備

2.1 AGORA

ゴール指向要求分析法では，顧客やユーザの要求を達成すべきゴールととらえ，それを AND-OR 分解し，より具体的なサブゴールの集合へと詳細化していく．システムとして実現する際には選択をする必要がある．分析結果は，ゴールグラフで表現される．AGORA[4),5)] では, ゴールグラフに属性，属性値をつけることができ, それを属性つきゴールグラフと呼ぶ. 属性, 属性値として, ゴール達成へサブゴールがどれだけ貢献するかの度合いを示す貢献度, 各ステークホルダがそのゴールが達成されるのがどれぐらい好ましいかの度合いを示す満足度行列, UML のステレオタイプのようなゴールの内容の意味的なカテゴリを示すセマンティックタグなどを扱うことができる. さらに, たとえば, ゴールを達成するためのコストなど, ゴールグラフ特有の属性とその計算式を定義できる. 本稿では, 満足度行列とセマンティックタグの 2 つを用いる.

図 1 は, 属性付きゴールグラフの例である. この例では, 病院受け付け業務の要求を分析している. 緑のゴールは, ステークホルダ間で採用するという合意が取られた最終ゴールを示す. この例では, N54 の “セキュリティが万全” というゴールに対し, N54 がセキュリティに関するものであることを示すように “Security” というセマンティックタグをつけている. 分析者は, ツールで標準的に提供されるセマンティックタグ以外に, 必要に応じて, その領域特有のものを追加することもできる.

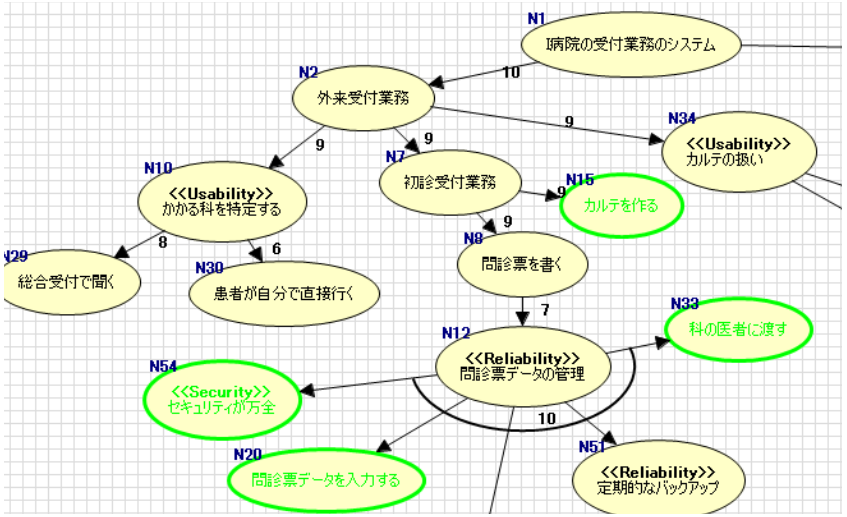


図 1　属性つきゴールグラフの例

図 2 は, N54 のゴールに付与された満足度行列であり, N54 に対するステークホルダ全員の値をまとめた結果を表している. 満足度行列には, 各ステークホルダがそのゴールが達成されるのがどれぐらい好ましいか, 他のステークホルダにとってどれぐらい好ましいと思われるかの度合いを –10〜10 で, 好ましい場合に, 正の値, 好ましくない場合は, 負の値を入力する. 図 2(b) のそれぞれの行, ユーザ (User), 管理者 (Administrator), 開発者 (Developer) はそれぞれのステークホルダが付けた点数を示す. この例では, N54 のゴールに対して, ユーザは, 自分にとっては，好ましくないものとして, –5 をつけ, 管理者にとっては好ましいものであると予想して, 8 をつけ, 開発者にとっては関心に無いものであろうと予想して 0 を付けたことを示す. 同様に他の 2 人のステークホルダも採点を行う．本稿では, 自分にとっての満足度である対角要素のみを扱う.

2.2 ステークホルダの関心事

使用性, 信頼性, 効率性等のシステム品質は, システム開発において一般的にステークホル

© 2011 Information Processing Society of Japan

Evaluator¥-ee	User	Administrator	Developer
User	-5	8	0
Administrator	8	9	0
Developer	8	8	

図 2 N54 の満足度行列

ダにとって, 重要な関心事である.

提案ツールでは, 次の 8 種類を品質特性をゴールのセマンティックタグに割りあてる関心事として標準的に提供している.

- **機能性 (Functionality)**: 目的から求められる必要な機能が実装されて度合
- **信頼性 (Reliability)**: 機能が正常動作続ける度合
- **使用性 (Usability)**: 分かりやすさ, 使いやすさ
- **効率性 (Efficiency)**: 目的達成のために使用する資源の度合
- **保守性 (Maintainability)**: 保守作業の容易性の度合
- **移植性 (Portability)**: 別環境へ移した際そのまま動作する度合
- **セキュリティ(Security)**: 不正アクセスからデータを守り, 悪意ある操作に対する耐性の度合
- **コスト (Cost)**: システムを開発する費用や手間

機能性から移植性までの 6 種類は, ISO/IEC9126-1[9] の外部・内部品質を表す特性を用いている. ISO/IEC9126-1 では, ソフトウェア品質を示す特性に, “外部・内部品質を表す特性” とそれにさらにブレイクダウンした “品質副特性” を定めている.

本ツールでは, ツールが標準的に提供する関心事以外にも, ステークホルダがその領域特有の関心事を定義することも可能であり, 独自に定義された関心事も標準的に提供する関心事と同じく扱う.

セキュリティは, ISO/IEC9126-1 では信頼性の一部であり, コストは, 品質としては分類されないが, これまでの経験で, ユーザがしばしば用いたため標準的に提供することにした.

2.3 アンカーマップ

例えば, ユーザ, 管理者, 開発者, それぞれ 3 人の合わせて 9 人のステークホルダの集合と Functionality, Reliability, Usability, Efficiency, Maintainability,Portability Security,Cost の 8 つの関心事からなる集合があったとする. 各ステークホルダと関心事の間に興味に基づく関係があるとする. ステークホルダと関心事を 2 種類のノード集合, ステークホルダと関心事の間の関係を枝と定義すると**図 3**(a) のようなグラフが定義できる. このように 2 種類のノードから構成されるグラフで, 異なる種類のノードの間に枝が存在し, 同じ種類のノードには枝が存在しないものを 2 部グラフと呼び, 図 3(a) のように描画したものを 2 階層レイアウトと呼ぶ.

アンカーマップとは 2 部グラフの 2 種類のノードの 1 種類に位置の制約を課して関係構造の把握を助ける描画手法である. アンカーマップは, ひとつの観点に制約を課して表示することによって, 複数の観点をもつノードの関係の理解を容易にすることができる. 制約を課すノードを**アンカーノード**, それ以外を**フリーノード**と呼ぶ.

アンカーマップでは, アンカーノードを一つの円の円周上に配置する. 円周上に配置した以外のノードは, フリーノードとして, 隣接するアンカーノードに関連して, スプリングアルゴリズムを用いて, 適切な位置に配置される. スプリングアルゴリズムは, 枝で結ばれたノード間には, 距離が離れれば離れる程, また枝に付与された値が大きければ大きい程, 大きな引力が働き, それ以外のノード間には, 距離が近ければ近い程大きな斥力が働くと考えて, 反復的にノードの位置を計算する. アンカーノードとフリーノードを入れ換えて, 同じようにアンカーマップを作成することもできる. すなわち, 一つの 2 部グラフから 2 種類のアンカーマップを作成することができる.

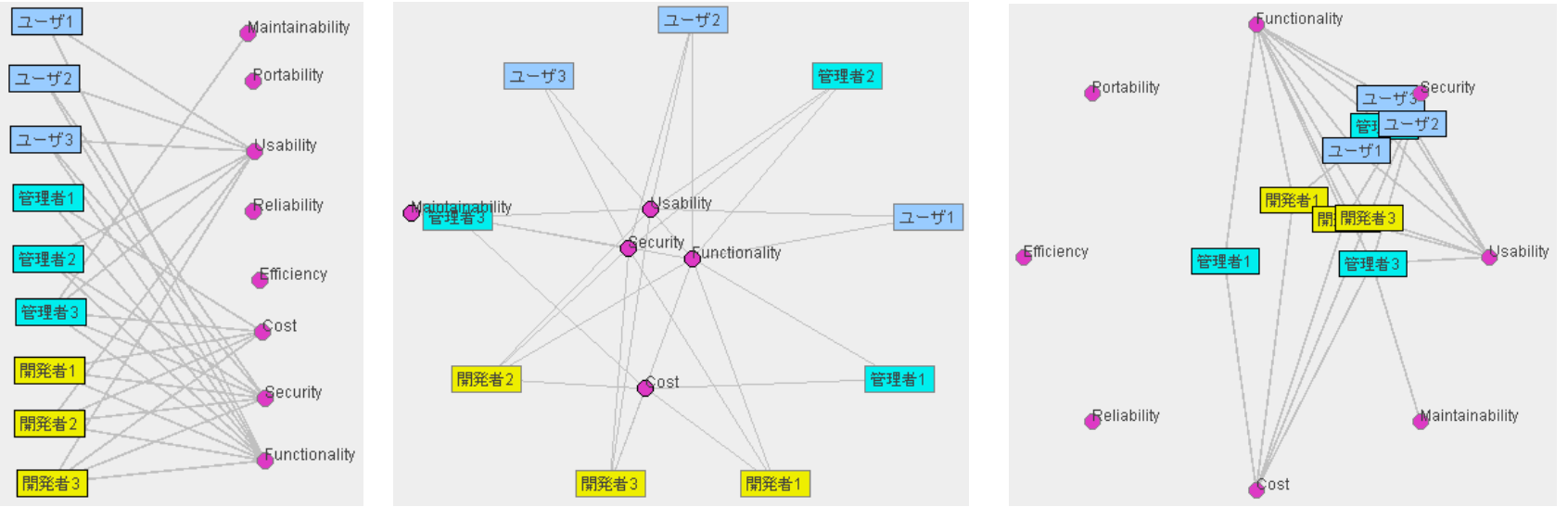
図 3(b), 図 3(c) は, 2 階層レイアウトで描画された図 3(a) と同じ 2 部グラフをアンカーマップで描画したものである. 例えば, 図 3(b) において, 管理者 3 と Maintainability がくっついているのは, Maintainability との間に枝を持つステークホルダのノードが, 管理者 3 だけしかなく, 管理者 3 と Maintainability の間の枝には大きな値が付与されていることを意味する. また, どのステークホルダとの間にも枝が無い Portability, Reliability, Efficiency の 3 つのノードは, 描画されない. 図 3(c) において, 開発者 3 の後ろに開発者 2 が一部, 重なってほぼ同じ位置に描画されていることは, 開発者 2 と開発者 3 が, 同じノード (Usability, Functionality, Cost) との間に共通して枝を持ち, それぞれの枝に付与される値も同じであることを意味する.

また, アンカーノードは, 同心円上に等距離に配置され, その位置は, 1) エッジの交差ができるだけ少なくなる, 2) 共通につながっているフリーノードが多い程近くになるなどの手法が提案されている[6].

3. ゴールグラフからのアンカーマップの生成

AGORA では, 各ステークホルダがそのゴールが達成されるのがどれぐらい好ましいかの度合いを 10〜−10 で入力する, 好ましい場合に, 正の値, 好ましくない場合は, 負の値, 無関心

© 2011 Information Processing Society of Japan



(a) 2 階層レイアウト　(b) アンカーマップ (アンカーノード: ステークホルダ, フリーノード: 関心事)　(c) アンカーマップ (アンカーノード: 関心事, フリーノード: ステークホルダ)

図 3　2 部グラフのレイアウト例

の場合は, 0 とする. したがって, 正負に関わらず, 絶対値が大きい場合にそのゴールに対して, 強い関心を示していると考えることができる. そこで, そのゴールに付与されているセマンティックタグによって表現されている関心事に対するステークホルダの関心の強さは, そのステークホルダの満足度の絶対値とする.

全てのゴールは, 関心事毎に集計される. ステークホルダと関心事の間の枝に付与する値は 関心事毎に集計されたゴールに付与されたステークホルダの満足度の絶対値の平均とする. 例えば, 図 1 に示したゴールグラフには, 信頼性のタグが付けられたゴールが 2 つ (N12 と N51) ある. このグラフに対して, ユーザというステークホルダが, 8 と 4 の満足度を付けたとすると, ユーザと信頼性の間の枝に付与する値は, 6(=(8+4)/2) となる. 同様にユーザと他の関心事, 他のステークホルダとそれぞれの関心事の間の値も計算する.

以上のように定義した, ステークホルダと関心事の関係を 2 部グラフで表現し, 図 3(a) のように図示した場合, だれも興味をもたない関心事を見つけることは容易にでき, その関心事に対するステークホルダが不足していることは容易に分かる. しかしながら, ある特定の関心事にある役割の人がだれも興味を持っていないことは容易には分からない. 例えば, この図では, ユーザの誰もコストに興味を持っていない. このまま続けるとユーザがコストを考えることなく要求を出し続けてしまい, 開発費が予算を大きく上回ってしまい, 後で機能を削減することが困難になるリスクがある. したがって, 要求獲得の早い時期からユーザにもある程度コストについて興味を持たせることが望ましい.

このような問題を解決するために, 提案ツールでは, ステークホルダと関心事の関係を可視化するためにアンカーマップを用いている.

枝で結ばれたノード間には, 距離が離れれば離れる程, また枝に付与された値が大きければ大きい程, 大きな引力が働くので, ステークホルダが, その関心事に強い関心があるとき, お互いにより近く配置される. 図 3(b) のノードの識別は, 図 3(a) よりも明解であり, フリーノード間の関係を容易に読み取ることができる. このように表現することによって, 強い関心を持つステークホルダがいない関心事の近くには, ステークホルダが配置されず, また誰も関心を持っていない場合には, どのステークホルダもその関心事につながらない. これに

© 2011 Information Processing Society of Japan

より, プロジェクトマネージャや分析者が関心事に対するステークホルダの不足を知ることができる.

アンカーノードの配置は, エッジの交差ができるだけ少なくなるようにしている. この配置アルゴリズムによって, 同じアンカーノードに対して枝を持つフリーノードが, 近くに配置され, ユーザは, 関心事に対するステークホルダの偏りを知ることができる.

同様のアンカーマップは, AGORA のゴールグラフである必要は無く, ステークホルダとそのステークホルダのそれぞれの関心事への興味の大きさが得られれば良い. 例えば, User Requirements Notation[10] のゴール指向要求記述言語 (GRN) の部分から同様のデータを作成して, 適用することが考えられる.

4. 実装と利用事例

4.1 ツールに対する要件

本ツールは, ソフトウェア開発会議において, そこまでに行なった要求獲得の成果物であるゴールグラフからステークホルダと関心事の関係がわかることが必要とされる. ある関心事に対して, 必要な知識を持っている人が含まれていなければ, 該当する人をステークホルダとして加え, ある関心事ばかり詳細化されるようであれば, 他の関心事に詳しい人をステークホルダに加えるという使い方が想定される. そこで, ツールとしては, 次のような機能が必要になる.

(1) ソフトウェア開発会議において, 複数人のステークホルダによって, ゴールグラフを作成できる.
(2) ステークホルダは, 任意の時点で追加でき, 追加されたステークホルダも各ゴールに満足度をふることができる.
(3) 作成途中のゴールグラフに対するステークホルダと関心事のアンカーマップが作成でき, 各ステークホルダがそれを参照できる.

4.2 実　　装

本ツールは, Eclipse のプラグインとして実装された AGORA エディタから呼び出す外部ツール (Java Applet) として実装した. 前節の要件 1 と要件 2 は, 既存の AGORA ツールを利用した. 要件 3 を満たすために, アンカーマップとして表示するデータは, ゴールグラフのエディタに表示されるゴールグラフのゴールに付与された満足度行列とセマンティックタグから作られる. アンカーマップは, 3 章で示したやり方を用いて描画される.

ステークホルダをアンカーノードとする場合 (図 3(b)), 関心事をアンカーノードとする場合 (図 3(c)) のいずれにおいても, 四角がステークホルダを示し, 小さな円が関心事を示す. 四角は, ユーザ, 開発者, システム管理者のそれぞれの役割毎に色分けされる. 各ノードは, マウスによって移動することができ, 複数のノードが重なった場合に, 上になったノードを移動させて, 下になったノードの内容を見ることができる. フリーノードは, 移動させた後, マウスを離すと, スプリングの描画原理にしたがって, ゆっくり元の位置に戻る. ただし, アンカーノードは, 最初に位置が計算されて配置されるため, 描画後に移動されると元には戻らない.

4.3 利 用 事 例

図 1 に示した病院の受け付け業務のシステムのゴールグラフに対して, 9 人のステークホルダに満足度を振らせて, 得たアンカーマップが図 3(b) と図 3(c) である. ステークホルダは, ユーザ 1 からユーザ 3 の 3 人のユーザ, 開発者 1 から開発者 3 の 3 人の開発者, 管理者 1 から管理者 3 の 3 人のシステム管理者の 9 人で構成されている.

図 3(b) では, ステークホルダは, アンカーとして, 一つの円の円周上に等間隔で配置され, 関心事はフリーノードとして, 配置されている. この図では, セキュリティと機能性の 2 つの関心事が円の中心付近に配置されていることから, 多くのステークホルダが, この 2 つの関心事にほぼ等しく興味を持っていることが分かる. 他にも次のようなことが分かる.

- 3 人の開発者と管理者 1, 管理者 3 は, コストに興味を持っているが, ユーザはだれもコストに興味を示していない.
- 管理者 3 だけが, 保守性に興味を示している.

ユーザが誰もコストに興味を示していないこのプロジェクトでは, ユーザが思い付くままに要求を出して, 開発の優先順位をつけることが困難になることが懸念される.

一方, 図 3(c) では, 関心事がアンカーノードとして, 一つの円の円周上に配置され, ステークホルダがフリーノードとして配置される. この図では, 管理者 2 は, 3 人のユーザが重なって隠れた位置に描画されている. この図からは, 次のようなことが分かる.

- 全てのユーザは同じ関心事をもっている. 管理者 2 もユーザと同様である.
- 全ての開発者も同じ関心事を持っている. しかし, 開発者は, ユーザよりもコストに近いところの配置されており, ユーザとは異なり, コストにも関心を持っていることがわかる.
- 全ての管理者がそれぞれ異なる関心事に興味を示している.
- だれも移植性, 信頼性, 効率性に興味を示していない. これは, ゴールグラフでは, アプリケーションのビジネス領域が主に記述されており, インフラストラクチャーに付いては記述されていないことによる.

これらから得られる知見は, (1) システムのインフラストラクチャーについて, 詳しい人を

© 2011 Information Processing Society of Japan

表 1 実験に用いたゴールグラフ

システム	ノードの数	使われた関心事の数	ステークホルダの数
フィードリーダ (F)	29	4	9
病院の受け付け業務 (H)	36	5	9

プロジェクトにいれるべきである, (2) 管理者の関心は, 統一すべきである, ということを示唆している.

この事例によって, 提案ツールがステークホルダの偏りと不足を可視化できたと言える. また, その他のステークホルダ管理のための有益な情報が得られるとも言える.

5. 評 価 実 験

5.1 目　　的

評価実験によって, 提案ツールが, ステークホルダの偏りと不足を短時間で正確に既存の技術よりも発見するのに役立つことを示す.

実験は, ステークホルダの偏りと不足に関する質問に答えるという作業を用いて, 既存の手法と比較する. 比較は, 次のように行なった.

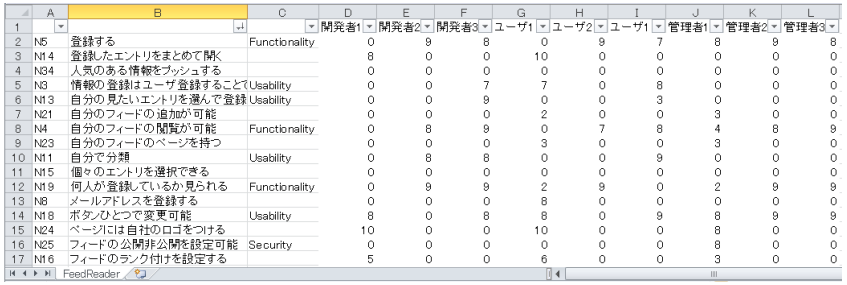
(1) 作業にかかった時間, 規定時間内に回答できた質問数, 回答できた質問に対する正解率
(2) 作業後に被験者が自分で予想した正解数
(3) 自由記述形式によるプロジェクトが持つリスクの記述量と内容
(4) 自由記述によるツールに対する意見

5.2 実 験 概 要

ゴールグラフ フィードリーダ (F) と病院の受け付け業務 (H) の支援システムのゴールグラフを用いた. それぞれのゴールグラフは, もともと 3 人 (管理者, エンドユーザ, 開発者) のステークホルダの役割を持った人によって, 作られたものであり, それぞれのゴールグラフは, **表 1** のような大きさである. 今回の実験のために, さらにそれぞれの役割の人 2 人ずつを割り当て, 満足度をふってもらった.

フィードリーダには, 効率性, 機能性, 使用性, セキュリティの 4 種類のタグが割り振られ, 受け付けシステムには, 機能性, 保守性, 信頼性, 使用性, セキュリティのタグが割り当てられた.

比較対象 **図 4** に示すような各ゴールに対して, 割り当てられた関心事と, そのゴールに対するステークホルダの満足度の一覧表を用いた. この表は, 第 1 列からゴールの ID, ゴール記述, セマンティックタグと続き, 最後に各ステークホルダの満足度で構成される. 満足度は, 無関心の場合は 0 で, 絶対値が大きいほど, 強い関心を示していることを意味する. 表は, Excel で被験者に提供し, 各列をキーにした並べ換えを容易にできるようにし, 並び換えや Excel のその他の機能は自由に使って良いことにした.



	A	B	C	D	E	F	G	H	I	J	K	L
1				開発者1	開発者2	開発者3	ユーザ1	ユーザ2	ユーザ1	管理者1	管理者2	管理者3
2	N5	登録する	Functionality	0	9	8	0	9	7	8	9	8
3	N14	登録したエントリをまとめて開く		8	0	0	10	0	0	0	0	0
4	N34	人気のある情報をプッシュする		0	0	0	0	0	0	0	0	0
5	N3	情報の登録はユーザ登録することで	Usability	0	0	7	7	0	8	0	0	0
6	N13	自分の見たいエントリを選んで登録	Usability	0	0	9	0	0	3	0	0	0
7	N21	自分のフィードの追加が可能		0	0	0	2	0	0	3	0	0
8	N4	自分のフィードの閲覧が可能	Functionality	0	8	9	0	7	8	4	8	9
9	N23	自分のフィードのページを持つ		0	0	0	3	0	0	3	0	0
10	N11	自分で分類	Usability	0	8	8	0	0	9	0	0	0
11	N15	個々のエントリを選択できる		0	0	0	0	0	0	0	0	0
12	N19	何人が登録しているか見られる	Functionality	0	9	9	2	9	0	2	9	9
13	N8	メールアドレスを登録する		0	0	0	8	0	0	0	0	0
14	N18	ボタンひとつで変更可能	Usability	8	0	8	8	0	9	8	9	9
15	N24	ページには自社のロゴをつける		10	0	0	10	0	0	8	0	0
16	N25	フィードの公開非公開を設定可能	Security	0	0	0	0	0	0	8	0	0
17	N16	フィードのランク付けを設定する		5	0	0	6	0	0	3	0	0

FeedReader

図 4 比較実験用ステークホルダと関心事の一覧表:フィードリーダ

作業 (質問) 被験者に作業として, 回答させるために 17 の質問を用意した. これらの質問は, 4.3 節で示したようなステークホルダの不足や偏りを導き出すものになっており, 例えば, 次のようなものである.

- だれも興味を持っていない関心事はどれか?
- 一人のステークホルダだけが興味を示している関心事はどれか?
- 全てのステークホルダが興味を示している関心事はどれか?
- 全ての開発者が興味を示している関心事はどれか?

1 番目の質問では, 該当する関心時に対する興味を持つステークホルダの不足が導き出される. 2 番目の質問と 3 番目の質問への答えを組み合わせることで特定の関心事にステークホルダの関心が偏っていることが導き出される. さらに 4 番目の質問への回答を組み合わせることで, 役割毎 (ここでは開発者) の関心の偏りを導き出すことができる.

被験者 ソフトウェア工学を専攻している大学院学生, ソフトウェア工学の基礎的な教育は受けている情報工学科の大学 4 年生, 大学に所属するソフトウェア工学の研究者, 企業につとめるソフトウェア開発者, それぞれ 1 人ずつ 4 人の被験者で実験を行なった.

手順 各被験者には, 2 つのゴールグラフのうち, 片方に対しては Excel で作成した一覧表とゴールグラフを用いて, 各質問に回答させる. もう一方に対しては, 提案ツールを用いて回答させる. ゴールグラフに対するツールの利用と Excel 表の利用は, **表 2** のように割

© 2011 Information Processing Society of Japan

表 2 作業の割り当てと結果

被験者	作業対象	提案ツール/一覧表	完了数	正解数	予想	時間
大学院生	F	提案ツール	17	17	16	7m10s
	H	一覧表	17	17	14	8m36s
大学生	F	一覧表	17	15	12	13m37s
	H	提案ツール	17	17	15	5m58s
研究者	F	提案ツール	17	17	17	6m19s
	H	一覧表	17	17	15	17m53s
エンジニア	F	一覧表	17	17	10	13m11s
	H	提案ツール	17	17	13	10m35s

り当てた. 各質問には, 2 分の制限時間を設け, 2 分以内に質問に答えられなければ次の問題に答えるように指示した. 全体の作業時間は, 25 分とした. 被験者は, 各質問に制限時間一杯かかれば, 12 問に回答することになり, 時間が余れば 13 問目以降にも回答し, 最大で用意した 17 問全てに答える. 制限時間内に 17 問全てに回答し終えた場合は, 作業はそこで打ち切った. 実験の前に, Excel による一覧表と提案ツールの使い方に慣れるために練習を行なった. 練習には, 5 つのゴールと 3 人のステークホルダ, 4 種類の関心事からなる小さなゴールグラフを用意した. 被験者には, このゴールグラフを対象に本番と同様の問題に, 一覧表とツールを使って, それぞれ 10 問ずつ回答させた.

5.3 結果と考察

表 2 の 4 列目は, 制限時間内 (25 分) に回答できた質問の数を示し, 5 列目は, 正解数を示す. 最後の列は, 全問の回答にかかった時間を示す.

- **完了数**: 全ての被験者が, 両方のゴールグラフに対して, 全ての問題に対して, 制限時間である 2 分以内に回答した. そのため, 質問への回答数に対するツールの使用の有無による差は無かった.
- **正解数**: 1 人の被験者が表を用いた場合に, 2 つの質問に対して間違いがあったが, その他は全ての質問に正しく回答した. 全ての被験者が, 提案ツールを用いた場合も, 表を用いた場合も全ての質問に正しく回答した.
- **時間**: フィードリーダで提案ツールを使った場合は, 合計で 13 分 29 秒かかり, 表を使った場合には, 26 分 47 秒かかった. 病院の受け付けシステムについては, 提案ツールを使った場合には, 16 分 33 秒, 表を使った場合には, 26 分 29 秒かかった.
- **自己予想**: 全ての被験者が, Excel による一覧表よりも提案ツールを使った方が予想正解数が多かった.
- **自由記述形式によるプロジェクトのリスク**: 次のようなリスクがあげられた.
  - 保守性に対して興味を持っている人がいないため, システム稼働後の運用に関することが十分に検討されていないと考えられる.
  - 信頼性に対して, ユーザと管理者が興味を持っていないため, システム稼働後のトラブル発生に対する対応方法が十分検討されていないと考えられる.
  - 管理者 1 が他の管理者と違う関心事に興味を持っているので, 優先順位付けのときに, 意見が割れると考えられる.

  大学院性が, 一覧表を使った方が多くのリスクを上げたが, 他の 3 人は, いずれもツールを使った方が多くのリスクを上げた. 記述内容については, ツール利用の有無による顕著な違いは見られなかった.
- **ツールに対する感想**: 次のような感想が得られた.
  - ツールの方が, 興味を持っていない特性を見つけやすい
  - ツールの方が, 特性毎にまとめて傾向を見るのによい
  - 特性に共通して関心を持っている人を見つけやすい
  - ステークホルダ別々ではなく, 役割毎にまとめて表示されるとよい
  - ツールでは, 満足度の値を確認できない
  - ステークホルダをアンカーノードにした図と, 関心事をアンカーノードにした図の両方を同時に見たい
  - ノードが重なってみにくい

以上の結果は, 以下のことを示唆している. 全ての被験者が, 一覧表よりも提案ツールを使った方が予想正解数が多かったということは, 自信を持って回答したと考えられる. プロジェクトが持つリスクに関する記述は, 作業の後に行なわれたため, 作業を行なっている中で気づいたことが記述されたと考えている. ツールの使用の有無により内容に差が出なかったのは, 同じ作業を実施したためであると考えている. ツールを使った場合に記述量が多かったのは, 質問に対する回答が短時間で少ない負担で実施できたからだと考えている. 被験者の数は少ないが, 上記結果は, ツールが, ステークホルダの偏りや不足を同定するのに, 既存によく使われているステークホルダと課題 (ゴール) の対応表よりも短時間で, 正しくできるために役立つことが示されたと言える.

6. 関 連 研 究

要求分析に関連して, グラフ構造を用いた研究として Heim は, 要求間の関係を可視化する

© 2011 Information Processing Society of Japan

技術を提案している[11]. 本稿で扱う関心事は, ソフトウェアの非機能要求を定義するのにも用いられる. Rohleder は, 8 つの組み合わせルールと 7 つのリンクタイプによって, 非機能要求の影響をグラフィカルに表現する手法を提案している[12] が, これは, 非機能要求の機能要求に与える影響を表現する点が, 本稿で述べたツールとは異なる.

ステークホルダの同定に関して, Pouloudi らは, 4 つの原則を示唆し, それらの原則に基づいたマネジメントの領域に効き目があるアプローチを提案した[13]. Lyytinen らは, 情報システムの開発におけるステークホルダの同定のためのガイドラインを示した[14]. これらの原則を提案ツールからプロジェクトの潜在的なリスクを読み取るためのガイドラインとして使うことが考えられる. Sharp らは, システムに関わる全てのステークホルダを同定する技術を提案している[1]. 我々の提案ツールは, AGORA のゴールのセマンティックタグの代わりに Sharp 等の提案するモデルを用いることで, 統合することも考えられる. しかしながら, Sharp 等の技術は, 要求分析とは別の作業として行なう必要がある. 我々の提案ツールは, ゴールグラフのエディタと統合されており, 不足しているステークホルダは, 分析作業の途中で見つけることができる.

7. ま と め

我々は, システムの信頼性, コスト, 使用性等のシステム品質に関する重要な関心事をもれなく獲得するためにステークホルダの不足や偏りを発見することを支援するツールを提案している. 提案ツールでは, ゴール指向分析手法の一つである AGORA の成果物であるゴールグラフを対象に, 要求分析作業中にステークホルダとステークホルダの品質に関する関心事の関係をアンカーマップを使って可視化する.

本稿では, 事例によって, ステークホルダの偏りや不足を見つけることができることを示した. さらに得られた気づきから, プロジェクトの潜在的なリスクが得られる可能性があることも示した. また, ツールの実装と, その評価実験について, 少ないサンプルではあるが, 良好な結果が得られたことを述べた.

今後の課題として, 事例を増やし, さらにツールの有効性, 実効性を示すことが必要であると考えている. さらに評価実験のサンプルを増やし, さらに効果の実証を行ないたいと考えている.

謝辞 本稿執筆にあたり, 下書きから丁寧に読んで有益なコメントをくださった東工大情報工学科の井上渉氏に感謝します.

参 考 文 献

1) Sharp, H., Finkelstein, A. and Galal, G.: Stakeholder identification in the requirements engineering process, *Proc. 10th Int. Database and Expert Systems Applications Workshop*, pp. 387–391 (1999).
2) Finkelstein, A. and Dowell, J.: A Comedy of Errors: the London Ambulance Service case study, *Proc. 8th Int. Workshop on Software Specificaiton and Design*, pp.2–4 (1996).
3) PMBOK: A Guide To The Project Management Body Of Knowledge (2005).
4) Kaiya, H., Horai, H. and Saeki, M.: AGORA: Attributed Goal-Oriented Requirements Analysis Method, *Proc. Requirements Engineering Conference*, pp.13–22 (2002).
5) Kaiya, H., Shinbara, D., Kawano, J. and Saeki, M.: Improving the detection of requirements discordances among stakeholders, *Requirements Engineering*, Vol.10, No.4, pp.289–303 (2005).
6) Misue, K.: Anchored Maps: Visualization Techniques for Drawing Bipartite Graphs, *HCI (2)*, Lecture Notes in Computer Science, Vol.4551, pp.106–114 (2007).
7) Ugai, T. and Aoyama, K.: Organization Diagnosis Tools Based on Social Network Analysis, *HCI (8)*, Lecture Notes in Computer Science, Vol.5617, pp.181–189 (2009).
8) Ugai, T., Hayashi, S. and Saeki, M.: Visualizing stakeholder concerns with anchored map, *Proc. 5th Int. Requirements Engineering Visualization (REV) Workshop*, pp.20–24 (2010).
9) *ISO/IEC 9126-1:2001 . Software engineering – Product quality – Part 1:Quality model*, ISO/IEC (2001).
10) Amyot, D.: Intorduction to the User Requirements Notation: Learning by Example, *Computer Networks*, Vol.42(3), pp.285–301 (2003).
11) Heim, P., Lohmann, S., Lauenroth, K. and Ziegler, J.: Graph-based Visualization of Requirements Relationships, *Proc. 3rd Int. Requirements Engineering Visualization*, pp.51–55 (2008).
12) Rohleder, C.: Visualizing the Impact of non-Functional Requirements on Variants : A Case Study, *Proc. 3rd Int. Requirements Engineering Visualization*, pp.11–20 (2008).
13) Pouloudi, A. and Whitley, E.A.: Stakeholder identification in inter-organizational systems: gaining insights for drug use management systems, *European Journal of Information Systems*, pp.1–14 (1997).
14) Lyytinen, K. and Hirschheim, R.: Information systems failures —a survey and classification of the empirical literature, pp.257–309, Oxford University Press, Inc. (1987).

© 2011 Information Processing Society of Japan