

2W-03 校閲機能を有する要求記述支援環境の提案*

安富 大輔[†] 増田 英孝[†] 佐藤 洋介[‡] 西山 裕二[‡] 井上 樹^{††} 佐藤 啓太^{‡‡}
 東京電機大学工学部[†] 武蔵工業大学工学部[‡] (株)豆蔵^{††} (株)デンソー^{‡‡}

1 はじめに

要求分析では、クライアントの要求を文書化した要求仕様と開発チームのメンバーが要求仕様で使われている用語に一貫性を持たせるために用語辞書を必ず作成する [1][2]。ソフトウェアは要求仕様をもとにして開発が行われるため、要求仕様に曖昧な単語や用語辞書に定義されていない用語が使用されたり、意味の矛盾や仕様の抜けといった問題を含むと、次工程でそれらの問題を解決するために、手戻りが発生し余計なコストがかかることになる。また手戻りができないところまで開発が進んでしまうと、開発チームが仕様を決定することになり本来の要求とは違ったソフトウェアを開発する恐れがある。

しかし、現状では要求仕様内の問題解決にはチェックリストを用いた人間側に依存した確実性に欠けた解決方法を行っている [3]。

そこで本研究では、要求仕様の記述での問題をチェックし、問題点を指摘する校閲機能を有する要求記述支援環境 [4] を提案する。

2 要求仕様における問題

要求仕様には、通常自然言語が用いられる。そのため以下のような曖昧な単語、用語辞書にはない用語の使用や要求仕様作成者の不注意や経験不足による要求仕様の矛盾や仕様の抜けといった問題が発生する。

2.1 未定義な用語の使用

用語辞書に登録されていない未定義な用語を使用すると、その用語の意味が定義されていないため、記述された内容は意味をなさない。また開発チームのメンバー各々で、意味の一貫性がなくなるため異なった解釈をしてしまう可能性がある。

具体例を挙げると、販売管理システムにおいて販売の内容を記述するために、「販売台帳」という用語が

定義されていたとする。この状況で、用語辞書には定義されていない「売上台帳」という用語を使用した場合、「売上台帳」という用語は用語辞書に定義されていないため「売上台帳」という用語の定義を行わない限り、それは意味の無い記述となり開発チームのメンバー各々で異なった解釈をしてしまう。

また、「販売台帳」と同じ意味だとすると用語辞書と対応ができていないことになる。そのためこの問題を解決するために、開発工程の手戻りが発生する(図 1 参照)。

2.2 曖昧な単語の使用

要求仕様に曖昧な記述(「など」、「適切な」、「十分な」等)を含んでいると、要求仕様が一意に決定しない。そのため、曖昧な記述がなされた部分を明確にしなければならなくなる。また開発工程の手戻りができない場合、本来仕様を決定する権限のない開発チームがソフトウェアの仕様を決めることになる。

具体例を挙げると、販売管理システムにおいて「顧客番号として適切な値を入力する」という記述が要求仕様にあったとき、「適切な値」とは具体的にどのような値なのかを明確にしなければならず、開発工程の手戻りが発生する(図 1 参照)。

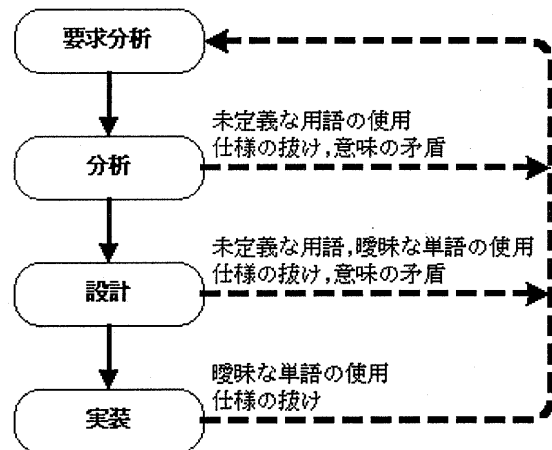


図 1: 各開発工程における手戻りの要因

*A Proposal of Requirement Description Support Environment has a Reviewal Function

[†]Daisuke YASUTOMI, [‡]Hidetaka MASUDA, [‡]Yosuke SATO,

[‡]Yuji NISHIYAMA, ^{††}Tatsuki INOUE, ^{‡‡}Keita SATO

[†]Tokyo Denki University, [‡]Musashi Institute of Technology,

^{††}mamezou co.,Ltd, ^{‡‡}Denso Corporation

2.3 仕様の抜けや意味の矛盾

要求仕様においては、仕様の抜けや意味の矛盾といった仕様の一貫性がとれなくなる問題も考えられる。

具体例を挙げると、用語辞書に定義されている「販売台帳」という用語が要求仕様に使われていないと、用語辞書にある「販売台帳」という用語は必要なのか、あるいは使われるべき用語であれば仕様の抜けである可能性があるため、開発工程の手戻りが発生する可能性がある（図1参照）。

3 提案する要求記述支援環境

上記の問題を含んだ要求仕様が、そのまま次工程に持ち込まれてしまうと開発チームのメンバーはクライアントと問題解決を行わなければならないとなり、開発工程の手戻りによりコストが増大する。また、開発チームが仕様を決定しなくてはならないためにクライアントの要求仕様にそったソフトウェアができない。そのため、要求仕様作成時に問題をできる限りチェックし排除しなくてはならない。

しかし、人間側に依存した問題解決では作成者の不注意や経験不足により問題解決の確実性に欠ける。

そこで、システムがあらかじめ用意した用語辞書や曖昧な単語の辞書を元に、要求仕様作成時に用語辞書に定義していない用語、曖昧な単語の使用、仕様の抜けや意味の矛盾をチェックし、作成者に誤りを通知する校閲機能を有する要求記述支援環境を提案する。

3.1 校閲機能

提案する機能は以下の機能である。

1. 用語辞書との整合性のチェック機能

要求仕様で使われている用語と用語辞書で定義されている用語との整合性をチェックし、システム利用者に通知する機能である。これにより、システム利用者が用語辞書に定義されていない用語を使用すると、システムはその旨を通知することにより、用語辞書に定義すべき用語なのか、用語辞書にある同じ意味の用語を使用すべきなのか把握できる。

2. 曖昧な単語のチェック機能

要求仕様の記述で曖昧な単語である「など」、「適切な」、「十分な」等の単語が使われた場合、その旨をシステム利用者に通知する機能である。これにより曖昧な単語をなくすことができ、曖昧な記

述を明確にするための開発工程の手戻りによるコストが削減できる。

3.2 開発支援機能

本システムでは、要求仕様からの開発支援のために、さらに以下の機能を提案する。

1. 記述の曖昧さ測定機能

曖昧な表現の出現頻度を評価することで、ドキュメントの正確さを定量的に測定する機能。

2. 辞書カスタマイズ機能

要求仕様で使用する辞書を、問題領域ごとに変更可能な機能。

3. クラス抽出機能

要求仕様からクラスとしてなりえる候補をシステムがあらかじめ抽出する機能。

4. パターン抽出機能

要求仕様からパターン [1] となりえる候補を抽出する機能。

5. コンポーネント通知機能

要求仕様に既存コンポーネントが適用できるような記述があれば、その旨をシステム利用者に通知する機能。

4 おわりに

上述した問題を解決するために校閲機能を有する要求記述支援環境を提案した。今後は、現在試作中の要求記述支援環境に各機能を実装し評価を行う予定である。

参考文献

- [1] Craig Larman(依田光江, 他訳) "実践 UML パターンによるオブジェクト指向設計ガイド", プレンティスホール出版, 1998
- [2] H.E. エリクソン, M. ベンカー (杉本宣男, 他監訳) "UML ガイドブック", トップラン, 1998
- [3] 萩本順三 "Developer's Rules of the Object-Oriented Process", NJK Corporation, 1999
<http://www.njk.co.jp/j/index.html>
- [4] 佐藤洋介, 西山裕二 他 "校閲機能を有する要求記述支援環境の構築", 第 61 回情報処理学会全国大会 2W-08, 2000