
ユーザ視点から機能要求・非機能要求を 導出するアプローチに関する検討

尾形 慎哉 亀和田 慧太 相沢 直人

株式会社U'eyes Design

目次

1. はじめに

-本研究の背景・目的-

2. 現状の課題

-機能要求・非機能要求を定義する難しさ-

3. 課題解決方針の提案

-ユーザ視点から機能要求・非機能要求を導出するアプローチ-

4. 有効性検証

5. まとめ

目次

1. はじめに

-本研究の背景・目的-

2. 現状の課題

-機能要求・非機能要求を定義する難しさ-

3. 課題解決方針の提案

-ユーザ視点から機能要求・非機能要求を導出するアプローチ-

4. 有効性検証

5. まとめ

システム開発の前提

システム開発プロジェクトにおける制約条件

Q:品質

自動車、飛行機、など

C:コスト

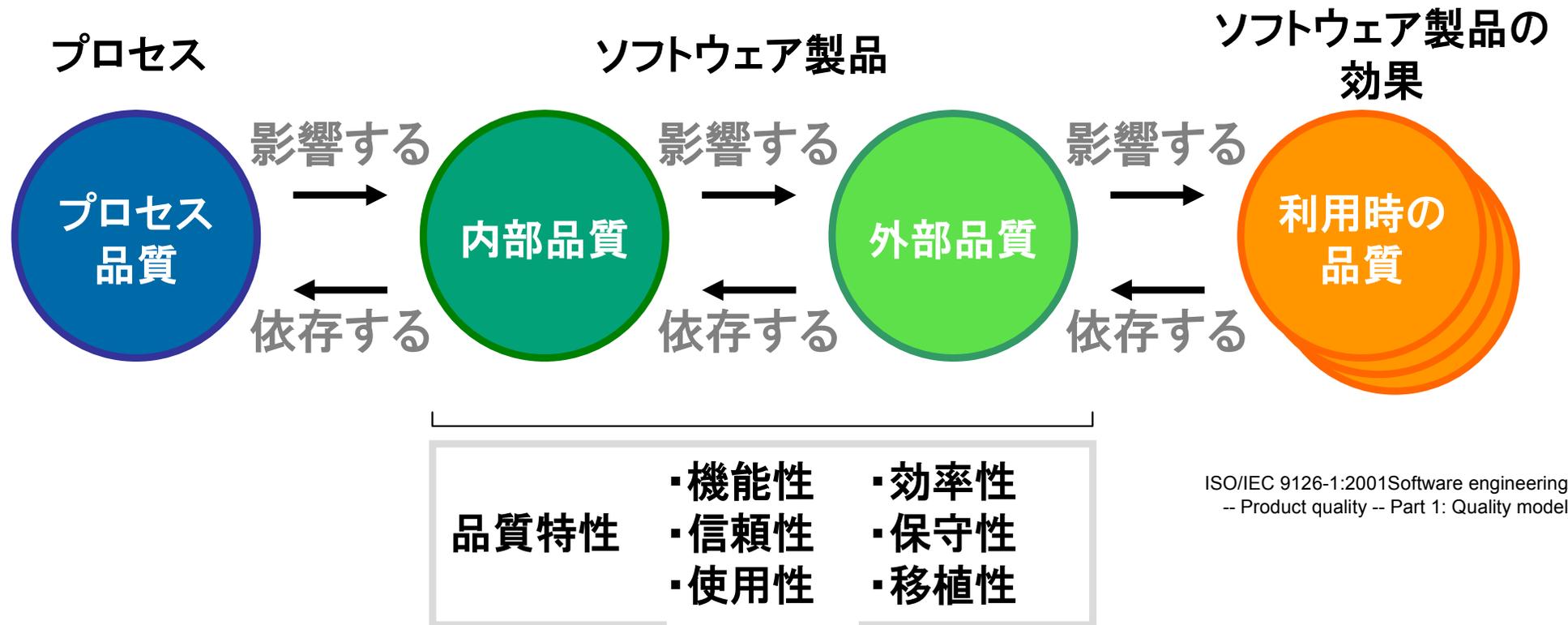
D:納期

イベントで使う情報システム、など

プロジェクトによって、QCDの優先度は変わってくるが、
システムの品質を高めることは不変の課題

ISO9126による4つの品質

- ・よい開発の仕組みは、よい品質の製品を生み出し、利用者が満足できる製品となる
- ・利用者が満足できる製品を作るためには、よい品質の製品を作る必要があり、そのための開発の仕組みが必要になる

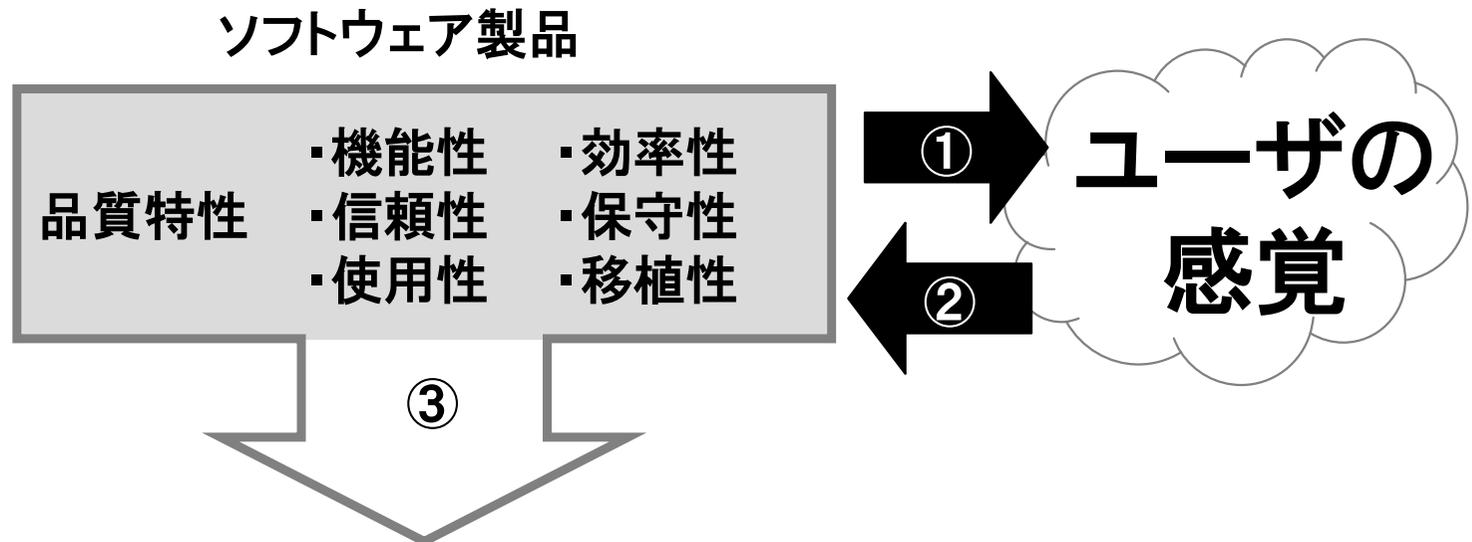


ソフトウェアとして、基準を満たしていればリリースは可能

品質に対する課題

品質の問題が発生する流れ

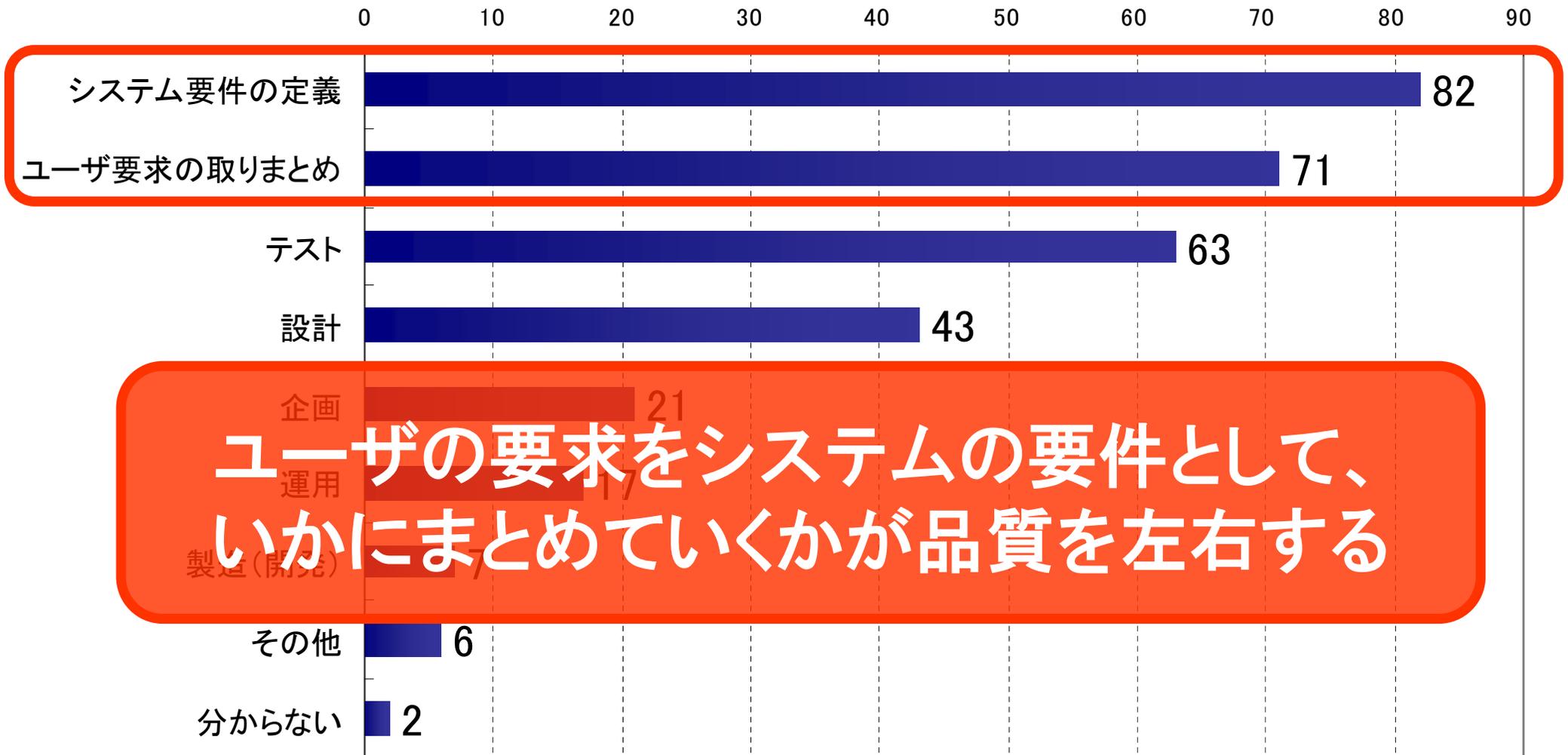
- ①ソフトウェア製品をリリース
- ②利用者からのクレーム
- ③不具合、リコールにつながる



**基準を満たしていても、
利用者の感覚に合っていないと不具合・リコール**

品質を高めるためのポイント

システムの品質を高めるために改善すべき工程について



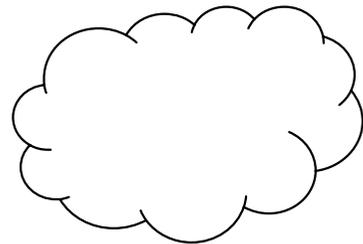
**ユーザの要求をシステムの要件として、
いかにまとめていくかが品質を左右する**

日経コンピュータ2011/1/20アンケート調査結果

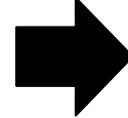
本研究の目的

課題解決のために

ユーザの感覚



何らかの
アプローチ



要件として定義する



要件として定義すること

- ・ユーザの要求する(欲しいと思う)機能が具備されていること→機能要求
- ・その機能の性能が、ユーザの感覚に合っていること→非機能要求

本研究の目的

ユーザの視点から機能要求・非機能要求を導出するアプローチの
提案とその有効性について検証する

目次

1. はじめに

-本研究の背景・目的-

2. 現状の課題

-機能要求・非機能要求を定義する難しさ-

3. 課題解決方針の提案

-ユーザ視点から機能要求・非機能要求を導出するアプローチ-

4. 有効性検証

5. まとめ

機能要求・非機能要求とは？（事例）

例：商業施設のバックヤードに設置されているエレベータ



開延長

荷物の運搬時などは、
扉が長く開いていて欲しい

ユーザ要求

扉を通常よりも長く開けて
おくことができるようにする

機能要求

対象：扉が開いている時間
ボタンを押下すると、
扉が40秒間開いていること

非機能要求

機能要求・非機能要求とは？

Robertson(2002)の定義

機能要件(要求)

- ・システムの機能性の仕様であり、
- ・システムが取らなければならない行動であり、
- ・システムの基本的な目的から引き出されるものであり、
- ・質ではない。

非機能要件(要求)

- ・ユースケースまたは機能要件によって表された作業の性格である。

利用時に問題が発生する要因

機能要求に問題がある場合

■機能要求

ユーザ要求から特定されたシステムに
具備しなければならない機能の要求



利用時→ ユーザの要求に対する
機能がない

機能要求の不備

例:扉を長く開けておくことができない

非機能要求に問題がある場合

■非機能要求

機能要求に対する性能・性格



利用時→ ユーザの利用状況に
合わない

**機能要求の性能に対する
ユーザ要求とのミスマッチ**

例:扉が延長されて開いている時間が短すぎる、長すぎる

目次

1. はじめに

-本研究の背景・目的-

2. 現状の課題

-機能要求・非機能要求を定義する難しさ-

3. 課題解決方針の提案

-ユーザ視点から機能要求・非機能要求を導出するアプローチ-

4. 有効性検証

5. まとめ

課題解決方針

• 機能要求の不備への対応

- 真のユーザ要求が捉えられていない
- ユーザがタスクを行うときの思考(期待・不安)の把握が不十分
- それを可視化できない
 - ユーザのタスクと思考を可視化する

• ユーザ要求と非機能要求のミスマッチへの対応

- 感覚は人によって異なるから一意に特定することは難しい
- 同じ性能でも、よいと思う人もいれば、悪いと思う人もいる
 - 極端なケースを考えて、許容範囲を徐々に絞って、性能値の範囲を特定する
 - ・どこに設定するかで製品の特徴が出る
 - ・同じ種類の製品でも、A社は起動が速い、B社はやや遅い、C社は遅すぎる、など
 - ・結果、企業の差別化要因につながる **A社は売り！ C社はクレーム！**

提案するアプローチ

STEP1. ユーザのタスクから「ユーザ要求」を導出する

STEP2. 導出したユーザ要求から機能要求を特定する
→機能要求への変換

STEP3. 機能要求に対する非機能要求の対象を特定する
→質、量、時間、速度

STEP4. 非機能要求の性能値の範囲を特定する
→性能の両極端の場合を想定する

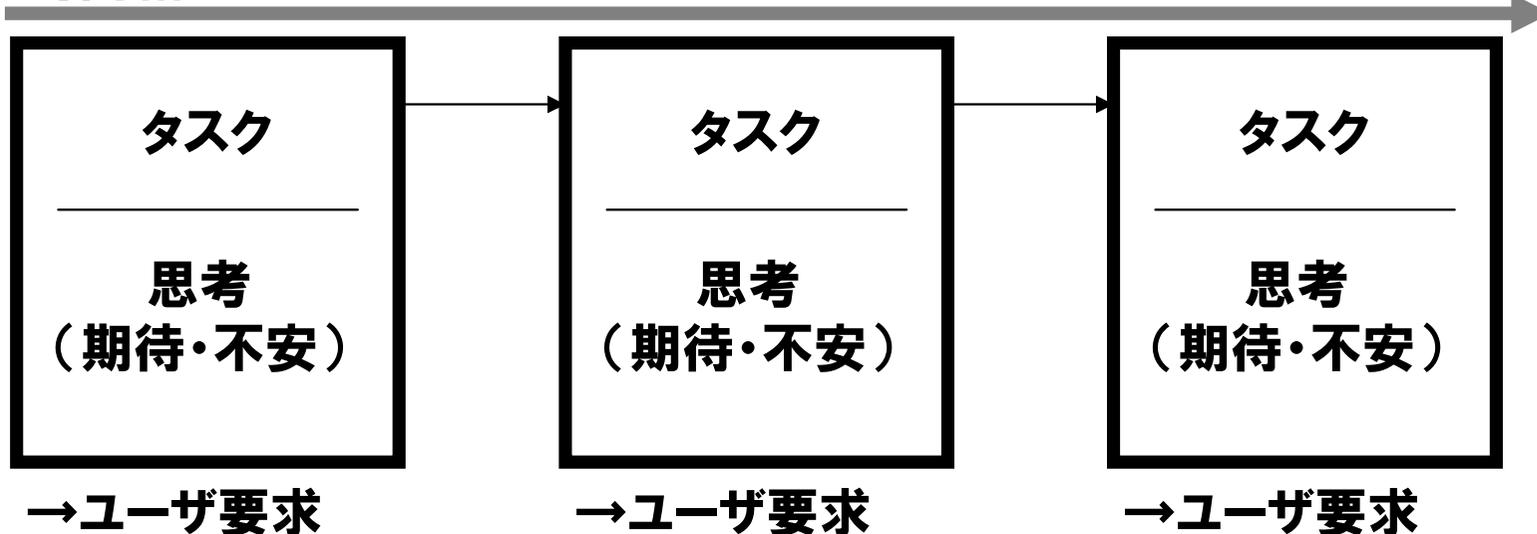
STEP1. ユーザのタスクから「ユーザ要求」を導出する

●ユーザのタスクと思考を可視化する

→ペルソナ・シナリオ法、フォトボーディング法、など

ユーザ視点フロー(タスクフロー)

時間軸



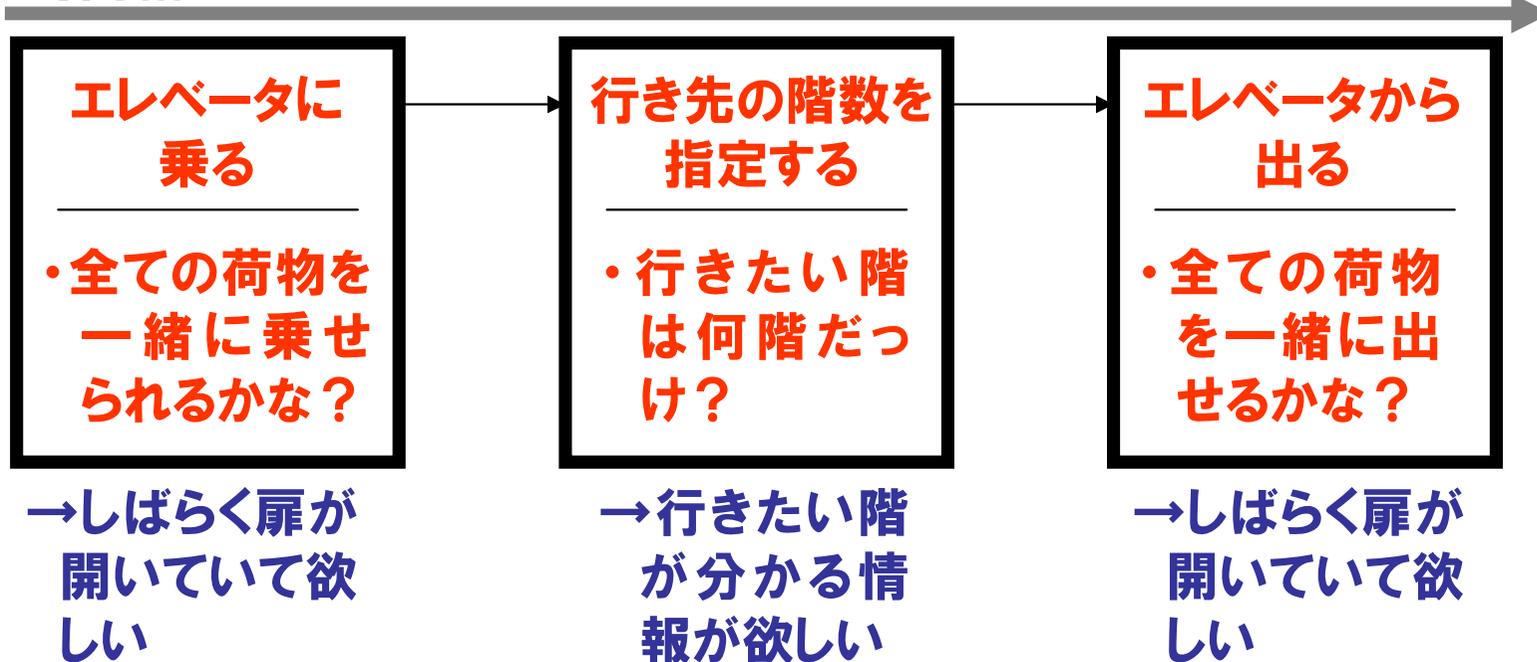
STEP1. ユーザのタスクから「ユーザ要求」を導出する

●ユーザのタスクと思考を可視化する

→ペルソナ・シナリオ法、フォトボーディング法、など

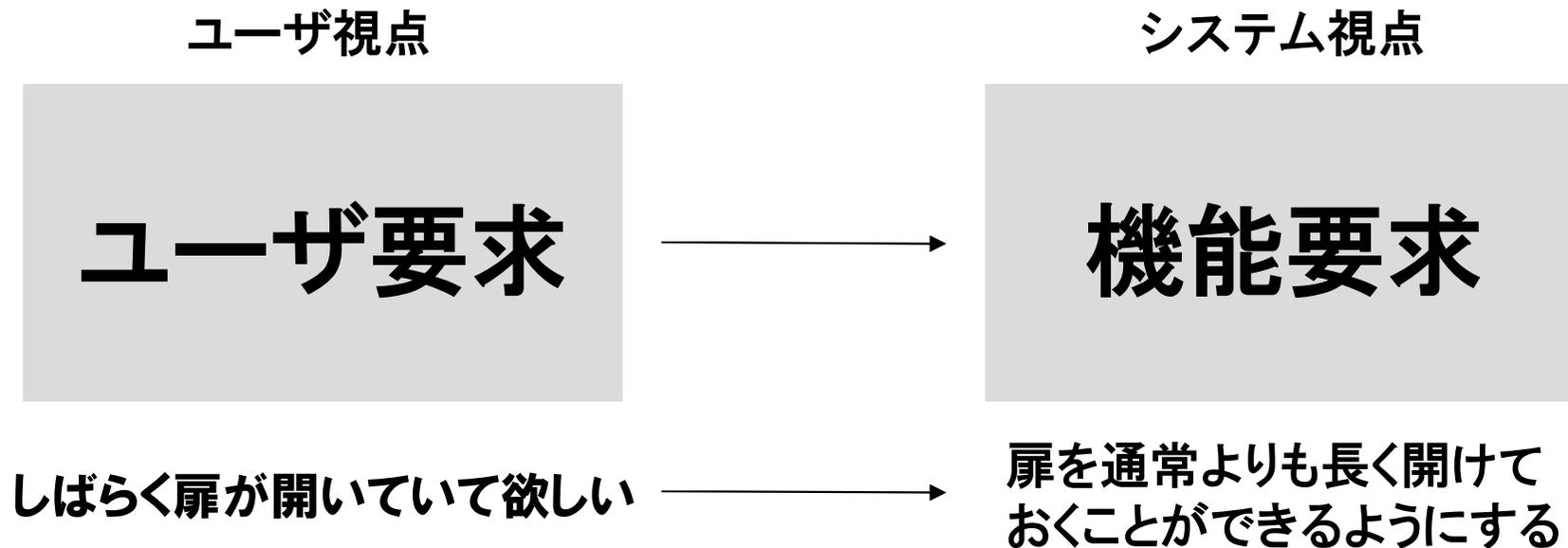
ユーザ視点フロー(タスクフロー)

時間軸



STEP2. 導出したユーザ要求から機能要求を特定する

●ユーザ要求を機能要求へ変換する



制約条件も考慮
※法規的な制約
※費用的な制約
※技術的な制約、など

STEP3. 機能要求に対する非機能要求の対象を特定する

●対象を質、量、時間、速度の観点から検討

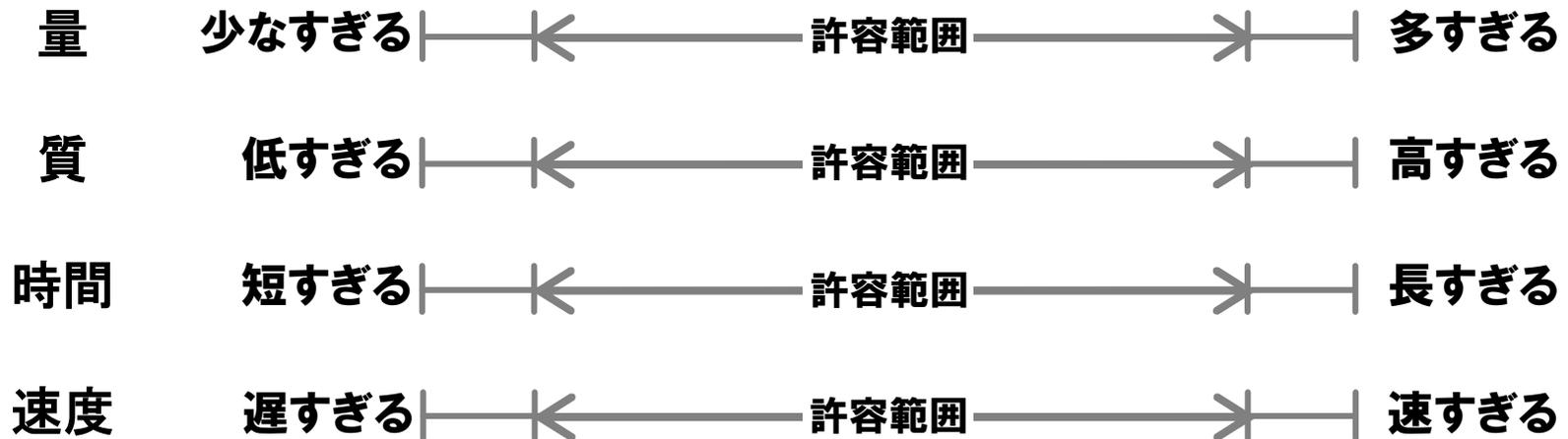


扉を通常よりも長く開けておくことができるようにする

扉の開いている時間

STEP4. 非機能要求の性能値の範囲を特定する

- 特定した観点で、極端な場合を想定した際に発生する影響を考慮して、性能値の許容範囲を検討する



時間	短すぎる	長すぎる
扉の開いている時間	荷物を出している途中で扉が閉まってしまう	他の階への移動待ち時間が長くかかってしまう

非機能要求検討ツール

量

量	対象				
	少なすぎると発生する問題				
	多すぎると発生する問題				
	非機能要求				
性能値	少なすぎる	←	許容範囲	→	多すぎる

質

質	対象				
	低すぎると発生する問題				
	高すぎると発生する問題				
	非機能要求				
性能値	低すぎる	←	許容範囲	→	高すぎる

時間

時間	対象				
	短すぎると発生する問題				
	長すぎると発生する問題				
	非機能要求				
性能値	短すぎる	←	許容範囲	→	長すぎる

速度

速度	対象				
	遅すぎると発生する問題				
	速すぎると発生する問題				
	非機能要求				
性能値	遅すぎる	←	許容範囲	→	速すぎる



量	対象
	少なすぎると発生する問題
	多すぎると発生する問題
	非機能要求

性能値 少なすぎる | ← 許容範囲 → | 多すぎる

質	対象
	低すぎると発生する問題
	高すぎると発生する問題
	非機能要求

性能値 低すぎる | ← 許容範囲 → | 高すぎる

時間	対象
	短すぎると発生する問題
	長すぎると発生する問題
	非機能要求

性能値 短すぎる | ← 許容範囲 → | 長すぎる

速度	対象
	遅すぎると発生する問題
	速すぎると発生する問題
	非機能要求

性能値 遅すぎる | ← 許容範囲 → | 速すぎる

時間

対象

扉の開いている時間

短すぎると発生する問題

荷物を出している途中で扉が閉まってしまう

長すぎると発生する問題

他の階への移動待ち時間が長くかかってしまう

非機能要求

**荷物搬入出の平均時間と、他の階で移動待ちの最大限許容できる時間を勘案して特定する。
→現状、40秒～3分の間とする**

性能値

短すぎる | ←



許容範囲 →



| 長すぎる

目次

1. はじめに

-本研究の背景・目的-

2. 現状の課題

-機能要求・非機能要求を定義する難しさ-

3. 課題解決方針の提案

-ユーザ視点から機能要求・非機能要求を導出するアプローチ-

4. 有効性検証

5. まとめ

検証するアプローチ

STEP1. ユーザのタスクから「ユーザ要求」を導出する

STEP2. 導出したユーザ要求から機能要求を特定する
→機能要求への変換

STEP3. 機能要求に対する非機能要求の対象を特定する
→質、量、時間、速度

STEP4. 非機能要求の性能値の範囲を特定する
→性能の両極端の場合を想定する

検証方法の概要

実験方法:

- ・システム開発担当者による機能要求・非機能要求の導出ワークショップ
→メールソフトの送信タスクにおけるユーザ要求、機能要求、非機能要求の導出

対象者:

- ・システム開発経験3年程度の2名1組

実施日時:

- ・2011年7月16日

実験の流れ:

- ・ユーザ要求、機能要求、非機能要求の導出を2回実施
 - ・1回目:特にやり方を指示せずブレスト形式で実施
 - ・2回目:本稿で提案しているアプローチ(ツール)を提供し、実施



結果

●獲得要求数

- ・ツールを使うことで、獲得できる要求が増えた。

	1回目 ツールなし	2回目 ツールあり	
ユーザ要求数	6	19	+13
機能要求数	14	17	+3
非機能要求数	4	11	+7

●要求の質（非機能要求）

- ・ツールなし:「ユーザがカスタマイズできること」など曖昧な記述がほとんど
- ・ツールあり:「全ての履歴を表示させること」など具体的な記述が見られた

結果

●参加者コメント

◎(ツールにより)非機能要求の観点を与えられたので、新たな

気づきは得られた

◎現場で情報を共有するにはよいツールかも知れない

×元々考えていたことを記述するだけ、という面もあった

考察

- **開発の要件定義段階で致命的な要件の検討抜け、漏れを防ぐアプローチとしては有効**
- **ツールを使うことで、新たな気づきを得られるきっかけになる**
想定外のことを想定する気づきにつながり、致命的な問題発見ツールとして使える可能性がある。
- **ただし、本ツールだけでは、非機能要求を特定するには不十分**
非機能要求の具体的な性能値までを特定するには、プロトタイピング手法などを組み合わせて検討していく必要がある。

目次

1. はじめに

-本研究の背景・目的-

2. 現状の課題

-機能要求・非機能要求を定義する難しさ-

3. 課題解決方針の提案

-ユーザ視点から機能要求・非機能要求を導出するアプローチ-

4. 有効性検証

5. まとめ

まとめ

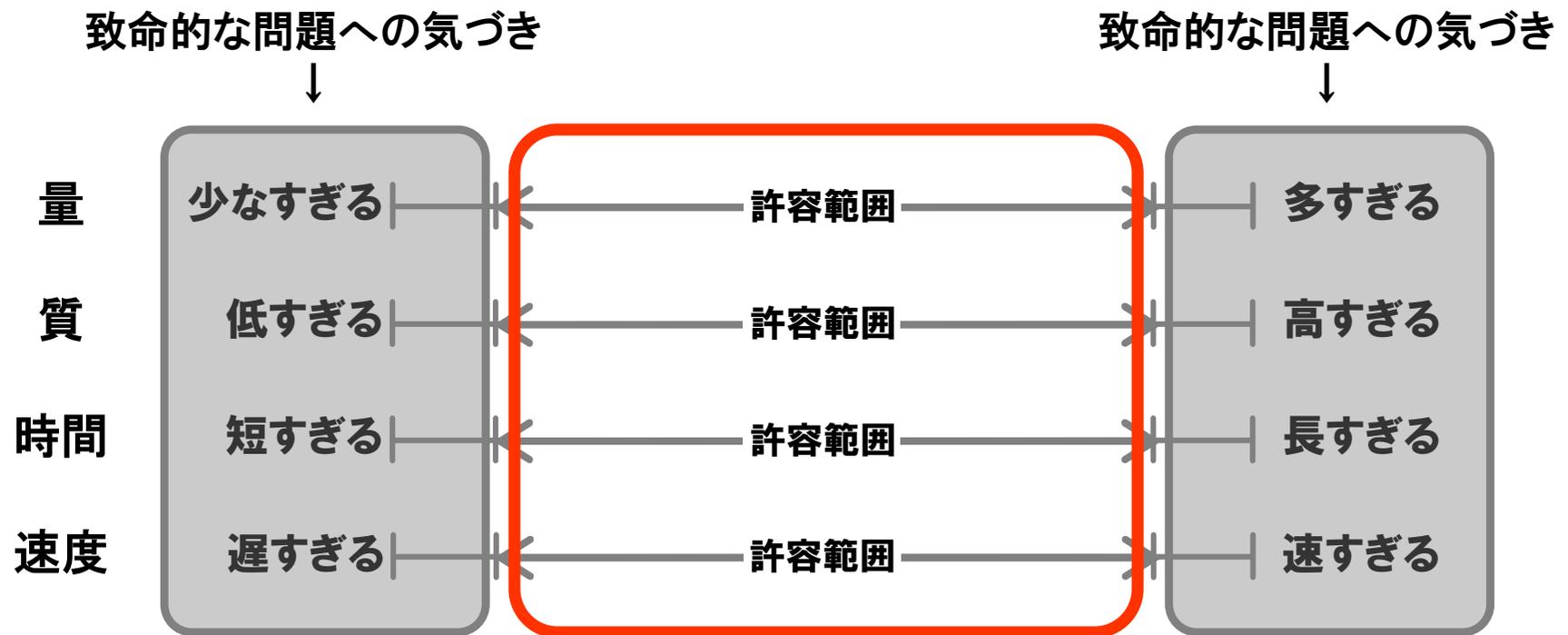
- ・ユーザ要求から機能要求・非機能要求を導出するアプローチについて検討を行った。

1. ユーザのタスクから「ユーザ要求」を導出する
2. 導出したユーザ要求から機能要求を特定する
→機能要求への変換
3. 機能要求に対する非機能要求の対象を特定する
→質、量、時間、速度
4. 非機能要求の性能値の範囲を特定する
→両極端の場合を考える

- ・結果として、要求についての新たな気づきを得るきっかけを与えることができた。
→非機能要求の致命的な問題に対する気づきを促すには有効。
- ・本アプローチで検討したツールのみでは、不十分。
→非機能要求の範囲の絞り込みまでは十分にできない。
→他の手法と組み合わせる必要がある。

今後の課題

・非機能要求としての性能値を特定するための検討



どう特定していくかが今後の検討課題

ご清聴ありがとうございました。

