

日本プロジェクトマネージメント協会 第118回 例会

システム開発のトラブルを防止する
要求仕様書とは
JUAS
2008.9.26

ユーザーとベンダーの両者がWIN-WINになるために

JUAS活動関係図

～ユーザーの要求が未来を切り拓く～

調査事業
・企業IT動向調査
・ソフトウェアメトリクス

**システムリファレンスマニュアル
(普及推進)**

政策研究会
・情報セキュリティ委員会
・CIO戦略フォーラム
・信頼性向上プロジェクト

会員研修会
・JUASアカデミー

・セキュリティセンター
プライバシーマーク審査
・UISSセンター
ユーザー版(IT人材育成)

フォーラム **会員活動**

- ・部門経営フォーラム(4)
- ・グループ会社経営フォーラム(2)
- ・IT企業TOPフォーラム(2)
- ・CIOフォーラム(3)
- (関西)・IT企業TOPフォーラム関西
- ・IT部門経営フォーラム関西
- ・IT企画推進フォーラム関西
- ・ITグループ会社フォーラム関西
- ・IT匠プラザ
- ・Lプラザ★

研究会

- ・IT戦略研究会
- ・人材育成研究部会
- ・情報共有研究会
- ・システム運用研究会
- ・企業リスクマネジメント研究会
- ・ユーザーIT活用研究会

研究プロジェクト

- ・システム開発・保守QCD向上プロジェクト
- ・OSS活用研究プロジェクト
- ・UVC研究プロジェクト●(User Vender Collaboration)
- ・要求工学研究プロジェクト●(RET:Requirements Engineering Technology)
- ・グローバルIT研究プロジェクト●
- ・ソフトウェア機能継承プロジェクト(旧ハードソフト寿命)
- ・サービス・サイエンス研究プロジェクト★
- ・高度IT化人材研究プロジェクト★
- ・SEの聞く力・話す力技術研究プロジェクト★
- ・ユーザー満足度研究プロジェクト●
- ・ユーザビリティ研究プロジェクト●

研修事業

- オープンセミナー
- オーダーメイド研修
- 教材開発・出版
- 海外研修

公開事業

JUASスクエア
(運営・企画委員会)

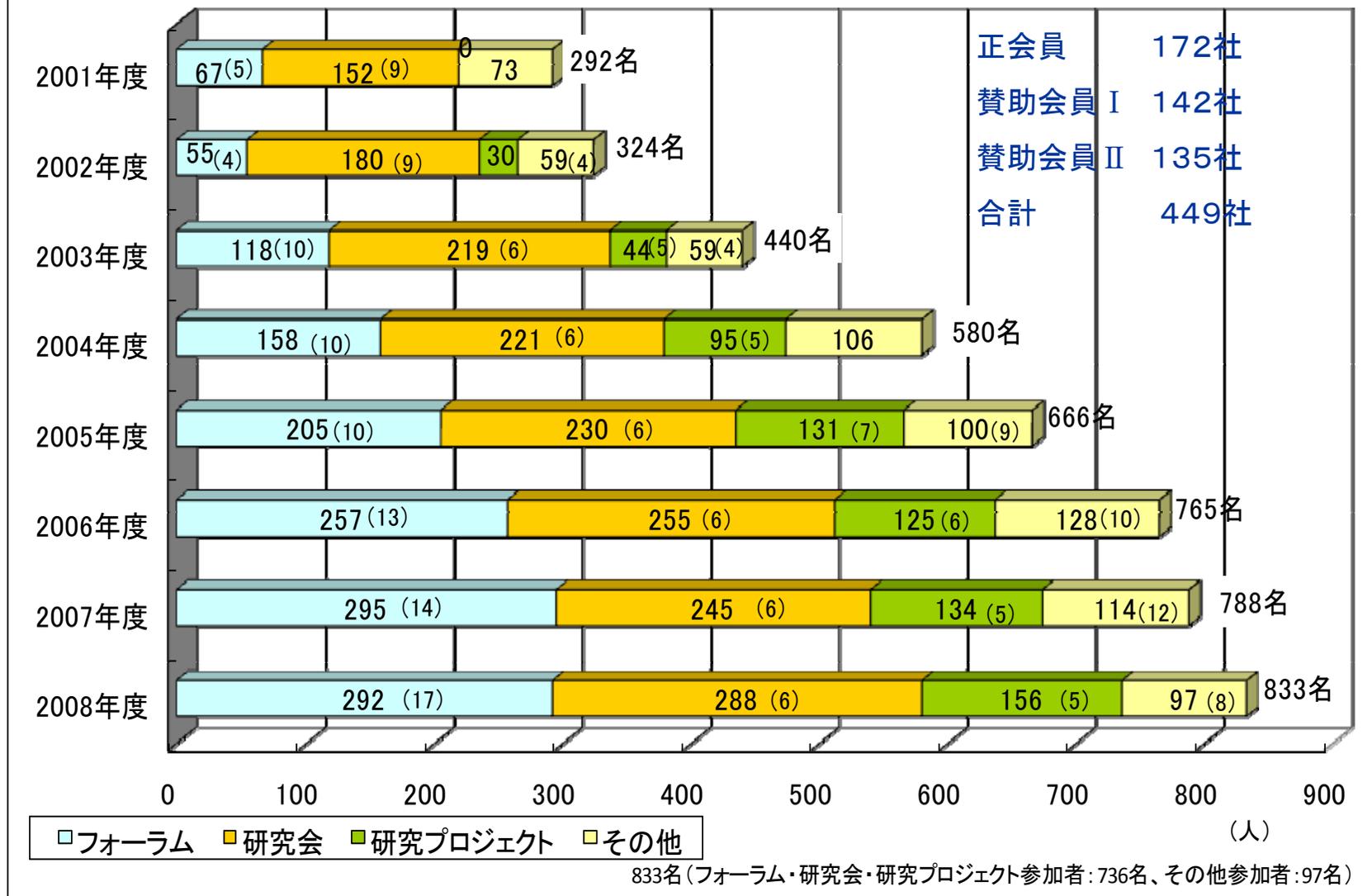
- ・京都スクエア
- ・ITガバナンス2008

●:2007年度以前に完了
★:2008年度新設

JUAS活動への参画人数の推移(2001～2008年度)

2008年9月15現在

JUAS活動参加人数の推移



要求仕様の明確化のための活動

順序	活動	実施年度
①	ユーザー満足度	2001,02
②	<u>エンドユーザーによるビジネスシステム定義の進め方</u>	2002,03
③	<u>ユーザーとベンダーが協力するリスクマネジメント</u>	2004,05
④	ソフトウェアメトリックス(開発・保守・運用の評価値)	2004~2008
⑤	ユーザビリティの基礎研究と分析	2004~06
⑥	<u>機能仕様の明確化(User Vender Collaboration1)</u>	2006
⑦	<u>非機能仕様の明確化(User Vender Collaboration2)</u>	2007
⑧	<u>ソフトウェア文章作法</u>	2004,05
⑨	<u>U字型開発法(プロジェクト管理)</u>	2004
⑩	サービス・サイエンス、重要インフラシステムの信頼性向上	2008
参考	システムレファレンスマニュアル1,2	2005,06

- ・情報を交換する場の提供から→情報を提供できるJUASへ
- ・従来の枠や発想にとらわれない活動
- ・基礎研究の重視

情報システムの上に業務システムがあり、その上にビジネスモデルがある

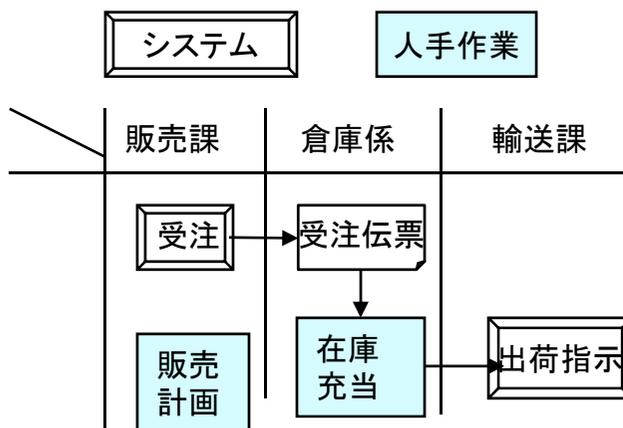
企画設計の流れ

③ 情報システム

要求仕様書→要件定義書→基本設計書→プログラム設計書

・システム機能構成表(機能概要)、システム関連図、業務処理ルール定義、画面帳票、データ項目定義、運用操作要件書

② 業務システム



- ・目標,目的,構想、組織役割
- ・工期、処理量、品質管理、
- ・データアイテム、コード
- ・業務フロー図(シーケンス図他)

現状業務を何を目玉にして変革するのか？効果をどこに求めるのかの視点が最重要(単なる業務標準化をしない)

① ビジネスモデル

・商品・サービス・顧客戦略 ・組織、設備、工場 ・要員数、
人材育成 ・資金

問題感知力・問題解決力

- * 何かおかしいと感じる力
- * もっと良いもの・方法がないかと考える力
- * 理想状態を想定できる力

現状把握・分析

問題点把握

理想像
想定

対策案
策定

設計
実装

運用
利活用

問題感知力

- ・ユーザー企業ではもっとも大切
- ・柔軟な理解力・発想力

独創力
柔軟性

問題解決力

- ・学校で習う内容
- ・たくましい実行力

活用力

- ・使いこなし力
- ・現場力

技術の伝承と進化

- ・感じ・考える力
- ・See→Touch→Measure→Analyze→Think→
- ・組織としての改善・改革風土

- ・改善技術 (IE, OR, QC, WD, KTなど)

育成阻害要因(解決可能)

- ・階層のフラット化・転職
- ・成果主義 ・長時間残業

読み・書き・考える力

要素技術
(これだけは最新技術の使える者が優先する)

プロジェクト管理技術

人間力

組織経営力

読み・書き・考える力

情報活用力

新ビジネス創造力

信頼性向上技術

人間力

組織経営力

プロダクト志向あつてのプロセス志向 (目標値を持った管理を)

貴方は商品・サービスを買う時に、製造元のプロセスを考えて買いますか？
それとも商品の質・価格・納期で買いますか？

良い商品・サービスを作る方法とは？

・①製造プロセスを確立すること(プロセス志向)

・* ISOのソフトウェア一部分 * CMM

・②最終商品の質(目標)を確保すること(プロダクト志向)

・* ハードウェア・・・6シグマ・・・欠陥商品は直ぐに取り替えます

・* ソフトウェア・・・バグがあるのは当たり前???

* Plan→Do→Check→Action

* 目標があるから、実績も評価でき改善アクションが見えてくる

* 貴社のシステム開発の品質？保守の品質？運用の品質？の

目標値とコストの関係は明確ですか？→ユーザとベンダー間の常識が必要

(優秀な商品・人が正当に評価される情報化社会を)

プロダクト志向とプロセス志向

1.目標管理と評価

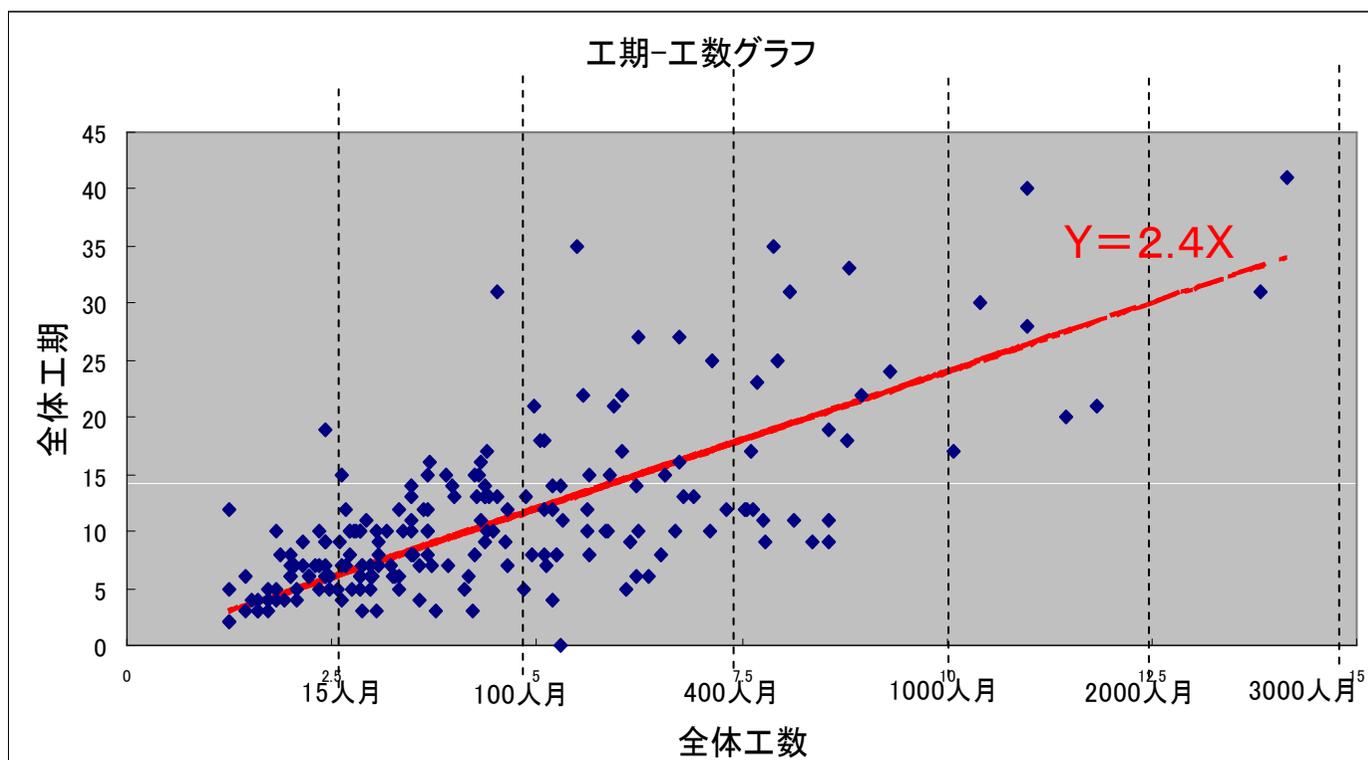
	ハードウェア (他の産業、機械工業・建設業)	ソフトウェア (情報産業)
商品の保証	規格や標準に規定されている 規格違反は法律違反となる 規格の種類は国別に多数あり	商品の品質特性の規定は存在するが 守るべき数値目標の規定はない
製造プロセス の規定	特に規定はない	開発フェーズ別になすべきActivityのガ イドは存在する。
不良品	欠陥品個数は6sigma以下の通念 不良品は即時取り替えが原則	バグはあるのが当たり前 不良品がまかり通る世界から徐々に許 されない世界へ
歴史	数千年の歴史を持つ	数十年の歴史
今後	無欠陥商品の追究 不良品は刑事責任を問われる	EASE(Empirical Approach to Software Engineering)などが出現
考え方	プロダクト志向	現在:プロセス志向 プロダクト志向あつてのプロセス志向

ソフトウェアにもプロダクト志向を

1.プロダクト志向をソフトウェア商品に持ち込む意味

	今まで	今後
商品品質	商品の質と値段の関係はない 機能性優先	「品質の良い商品は値段が高い」ことが社会通念になる 非機能性含めての評価
生産性	製造プロセスにおいて、高い生産性を出しても認められない。 しかし人により数倍異なることは常識	製造プロセスにおいて、高い生産性を出せば認められる
工期	工期保証を契約に盛り込むことは、提供者は回避しがち	条件があつての工期保証が成立
品質評価	良い商品を作る人・企業は評価されない	良い商品を作る人・企業は評価される
産業・人の評価	出来る人への作業負荷増大 若い人が魅力のある産業として認めていない	無茶な労働・残業の回避 魅力のある産業として優秀な若い人が集まる
管理方式	What to do/phase	What to do/phase+目標値の設定とその実績をフォロー PDCAが問われる

標準工期(適正工期)の考察(1)



- プロジェクト全体工数と、全体工期がともに記入されている**203プロジェクト**について、工数の3乗根と工期の関係をグラフ化し、回帰直線を引いた。
- 各企業においてこの式の何%までの短縮に耐えられるかは変わってくる

納期(工期)の評価尺度とアクション

●納期(工期)に関する問題(基本設計からカットオーバー迄)

	標準より長い工期	標準	25%工期短縮	25%以上工期短縮
工期の標準の考え方	金融等欠陥の発生を無くしたい品質重視のプロジェクトの場合	工数の立方根の 2.4 倍 (例:1000人月のプロジェクトは24 箇月)	・ユーザの要望 ・流通業のシステム化などに多い。	ユーザのやむを得ない外的事情で実施する場合 (対コンペ戦略、新商品の販売、株式の上場、企業の統合など)
スケジュールリングの対応策	十分なシステムテスト期間の確保	中日程計画の充実 (役割分担別WBS管理)	中日程計画の充実 (週間別管理)	小日程計画の充実 (日別管理)
その他の対応策	<ul style="list-style-type: none"> 品質重視のテスト計画書及びテストケースの緻密化 安定稼動のための分割立ち上げ等 	<ul style="list-style-type: none"> WBSによる総合計画と局面化開発 レビューの徹底 テストケース充実 コンバージョンデータのフル活用 確実な変更管理 	<ul style="list-style-type: none"> 同 左 + PGの選抜 *標準化の徹底と実力のある一括外注の採用。 システム範囲、対象の部分稼動 RAD+DOA 性能事前検証 変更管理の強化 	<ul style="list-style-type: none"> 同 左 + ベテランPMによる采配と会社あげでの協力及び監視 パート図での計画 ベストメンバー選出 クリーンルーム手法 二交代制の配置 顧客主体のテストチーム設置 パッケージの活用 部品の再利用 オープンな進捗情報管理

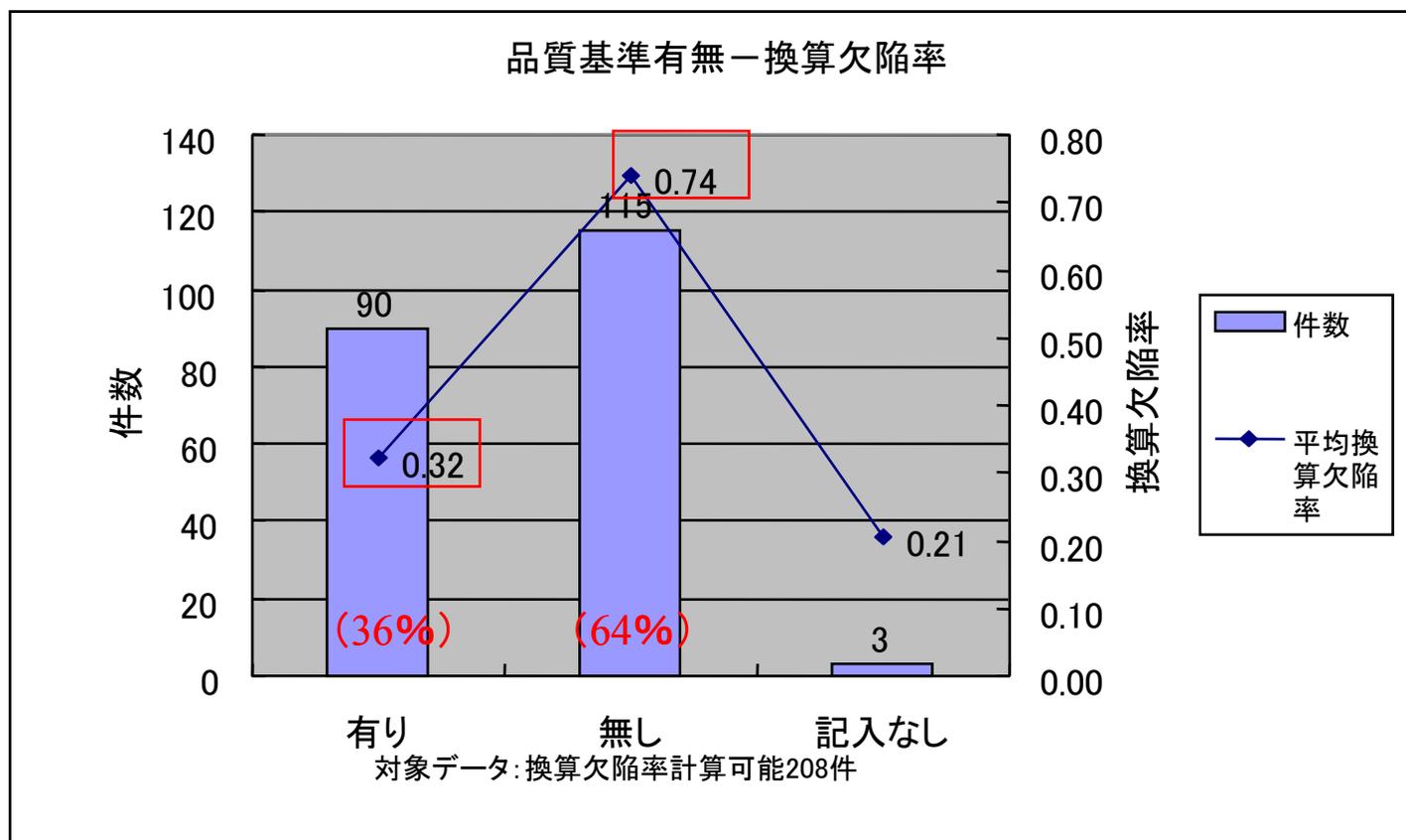
標準工期と実行工期の差(工期短縮率%)に着目してノウハウを蓄積する

品質基準の有無と欠陥率

品質目標を持っていたプロジェクトと目標が無いプロジェクトでは換算欠陥率において2.3倍の差があった。発注者は、発注時に品質目標を示すこと

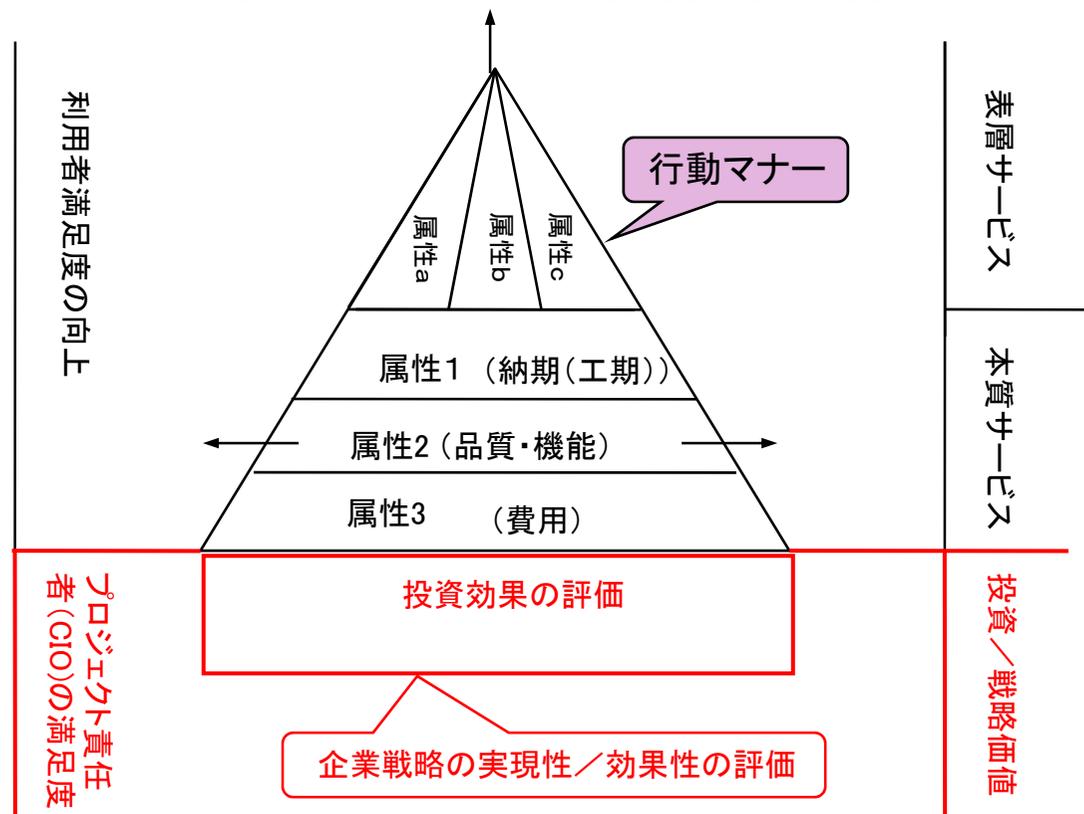
換算欠陥数 = 不具合大 × 2 + 不具合中 × 1 + 不具合小 × 0.5

換算欠陥率 = 換算欠陥数 / 工数 (顧客総合テスト～安定稼働の不具合数)



JUASのユーザ満足度コンセプト

JUASが考える、サービス充実度と満足度の関係



「ソフトウェア・サービスとは何か」を2008年度に「サービス・サイエンス・プロジェクト」を設置し、新しく研究中である。

欠陥率と顧客満足度の関係

欠陥率と顧客満足度(品質):50人月以上

小規模プロジェクトでは満足度が甘くなる可能性があるため、50人月以上のプロジェクトで計算した。縦軸は欠陥数/総工数

欠陥率		顧客満足度(品質)				計	満足率
		満足	やや不満	不満	未回答		
0	件数	6				6	100.0%
	平均	0.00				0	
0.25未満	件数	22	9		2	33	71.0%
	平均	0.09	0.11		0.11	0.10	
0.5未満	件数	8	6		2	16	57.1%
	平均	0.37	0.31		0.32	0.34	
1未満	件数	4	4	4		12	33.3%
	平均	0.67	0.77	0.64		0.69	
3未満	件数	6	4	2		12	50.0%
	平均	1.85	1.97	1.53		1.84	
3以上	件数	1	1			2	50.0%
	平均	3.05	4.35			3.70	
計	件数	47	24	6	4	81	61.0%
	平均	0.46	0.76	0.94	0.21	0.57	

- 欠陥率が1未満までは、品質が悪くなると満足率も下がっている。
- 欠陥率が1以上あっても迅速な対応で満足度を向上させている可能性がある。
- (このユーザ満足度の問題は2008年以降の「サービス・サイエンス・プロジェクト」でさらに追究してゆく)

プロジェクト開発の実態

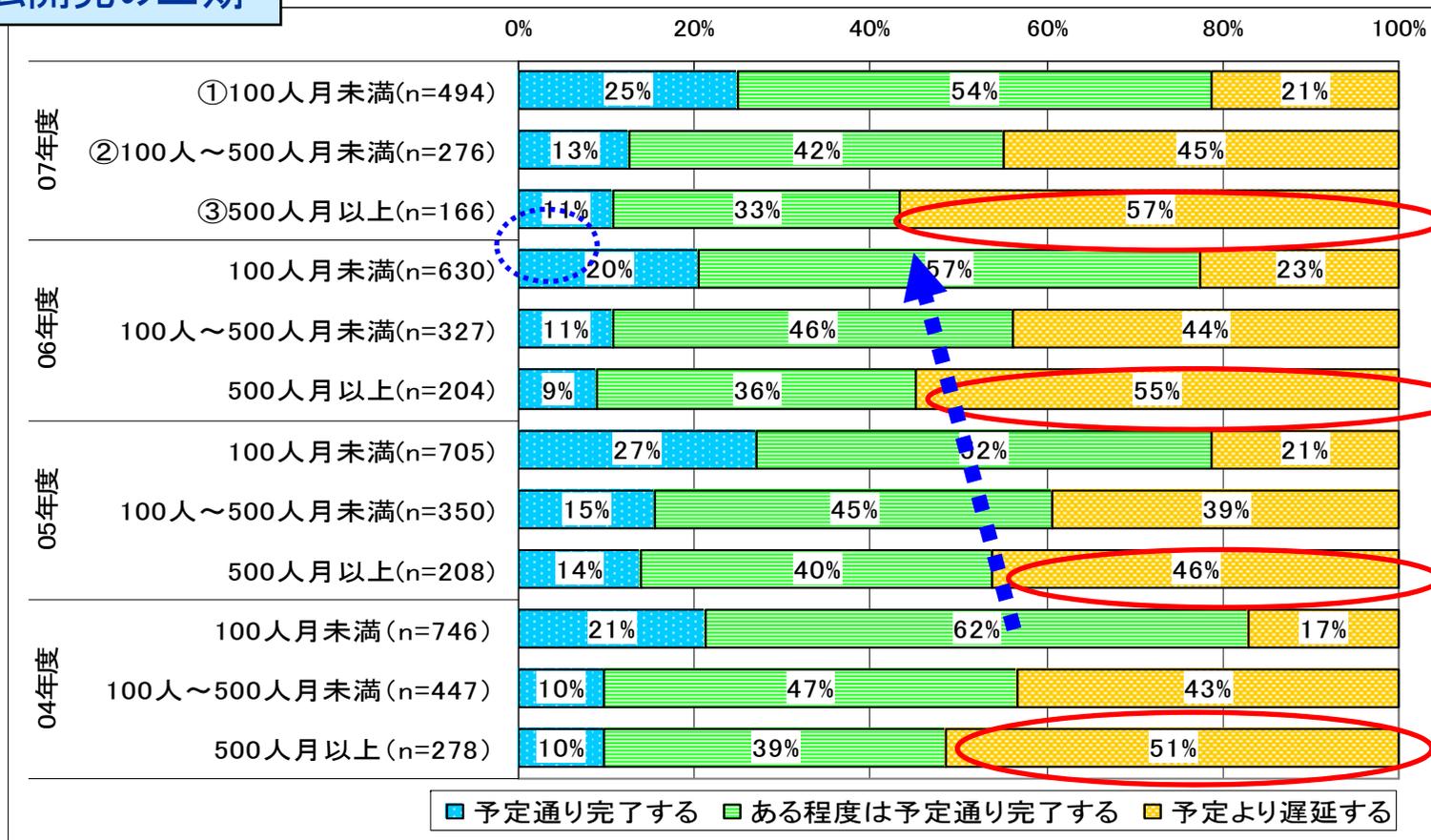
JUASの

- ・ソフトウェアメトリクス調査
- ・IT動向調査調査より

プロジェクトの実態

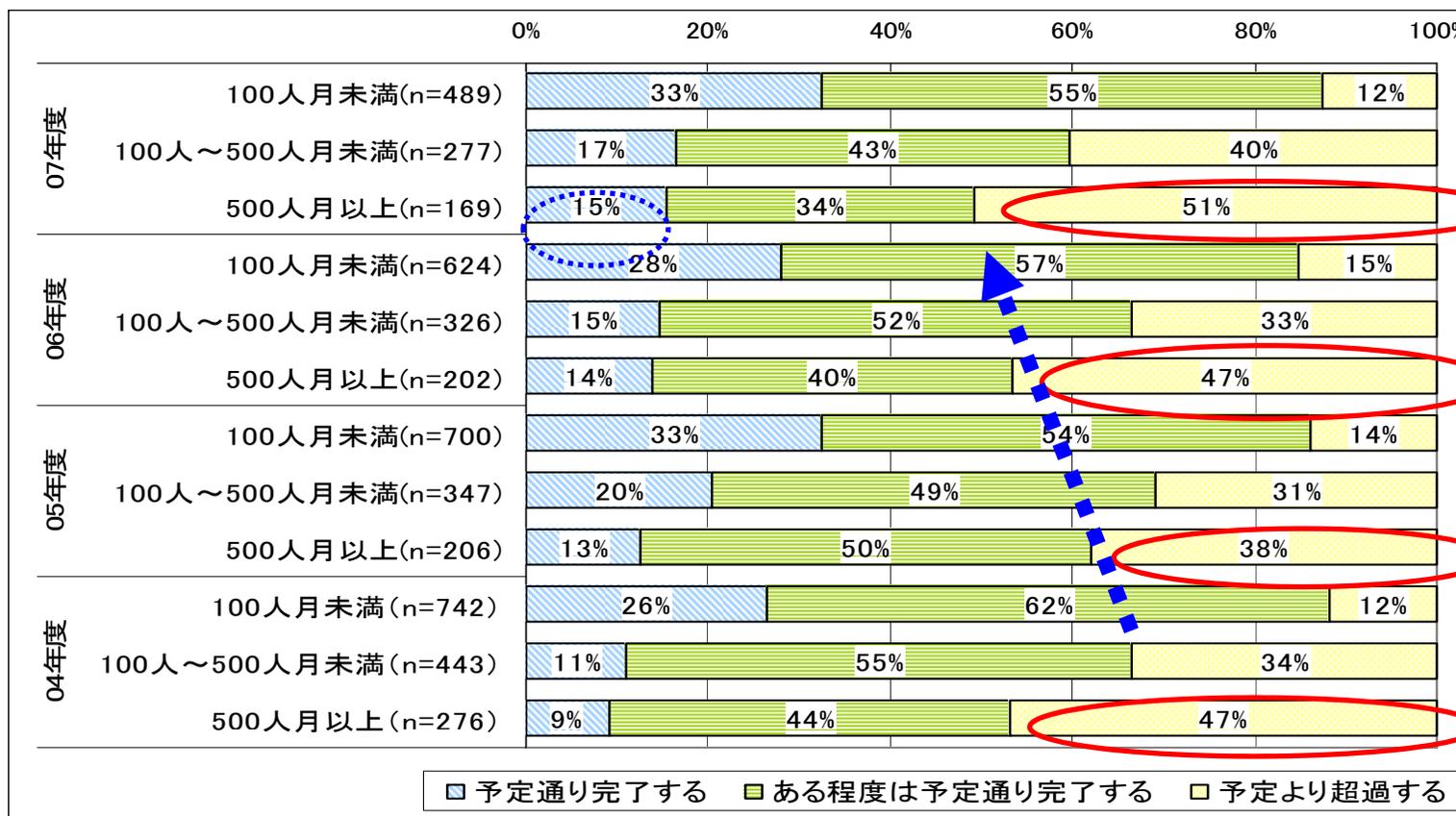
500人月以上の大規模プロジェクトでは相変わらず、6割近くのプロジェ
 クトで工期遅れが発生

システム開発の工期



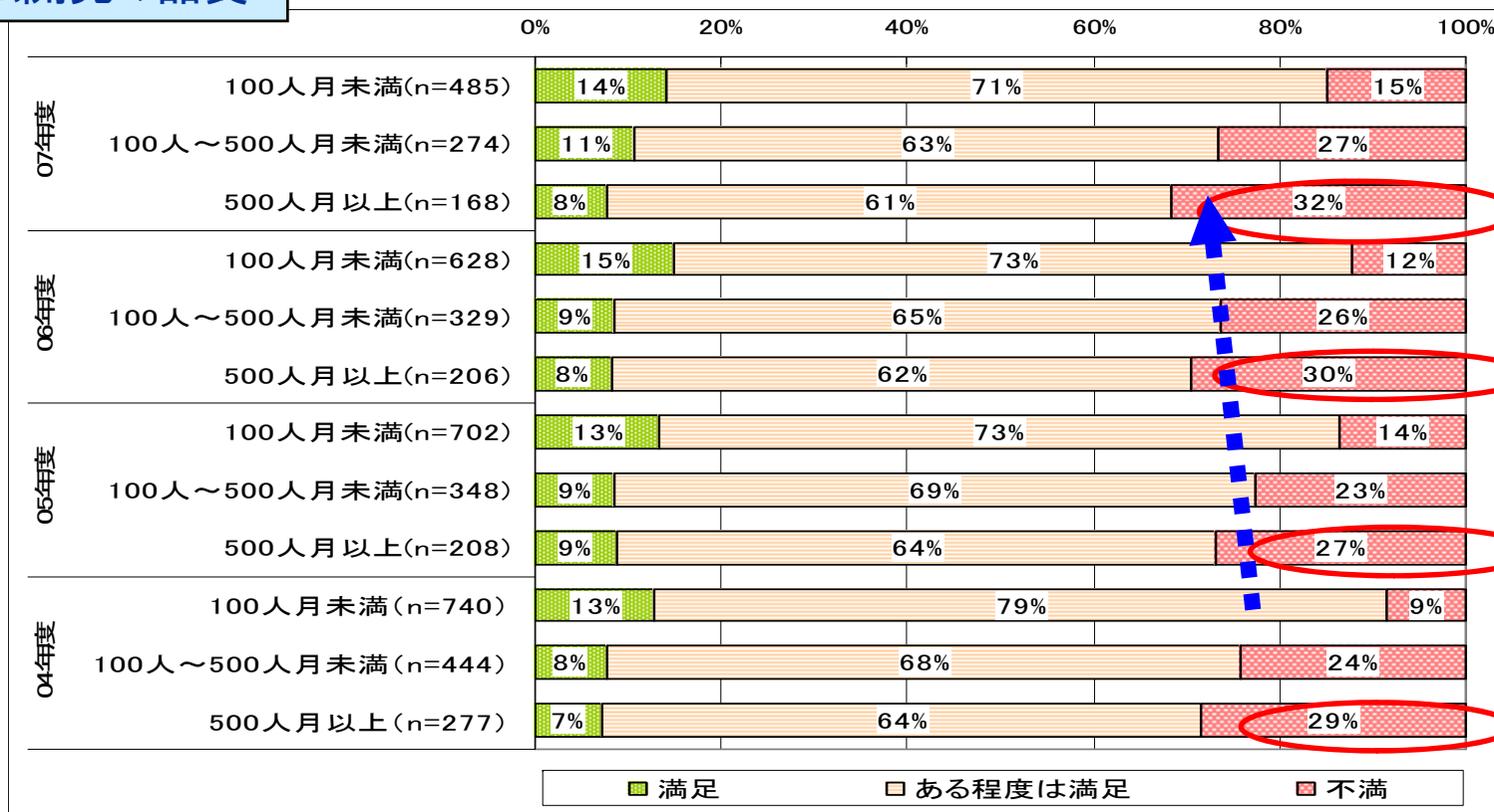
500人月以上の大規模プロジェクトの予算も 予算超過が増加して、「予定より予算超過」が過半数に

システム開発の予算



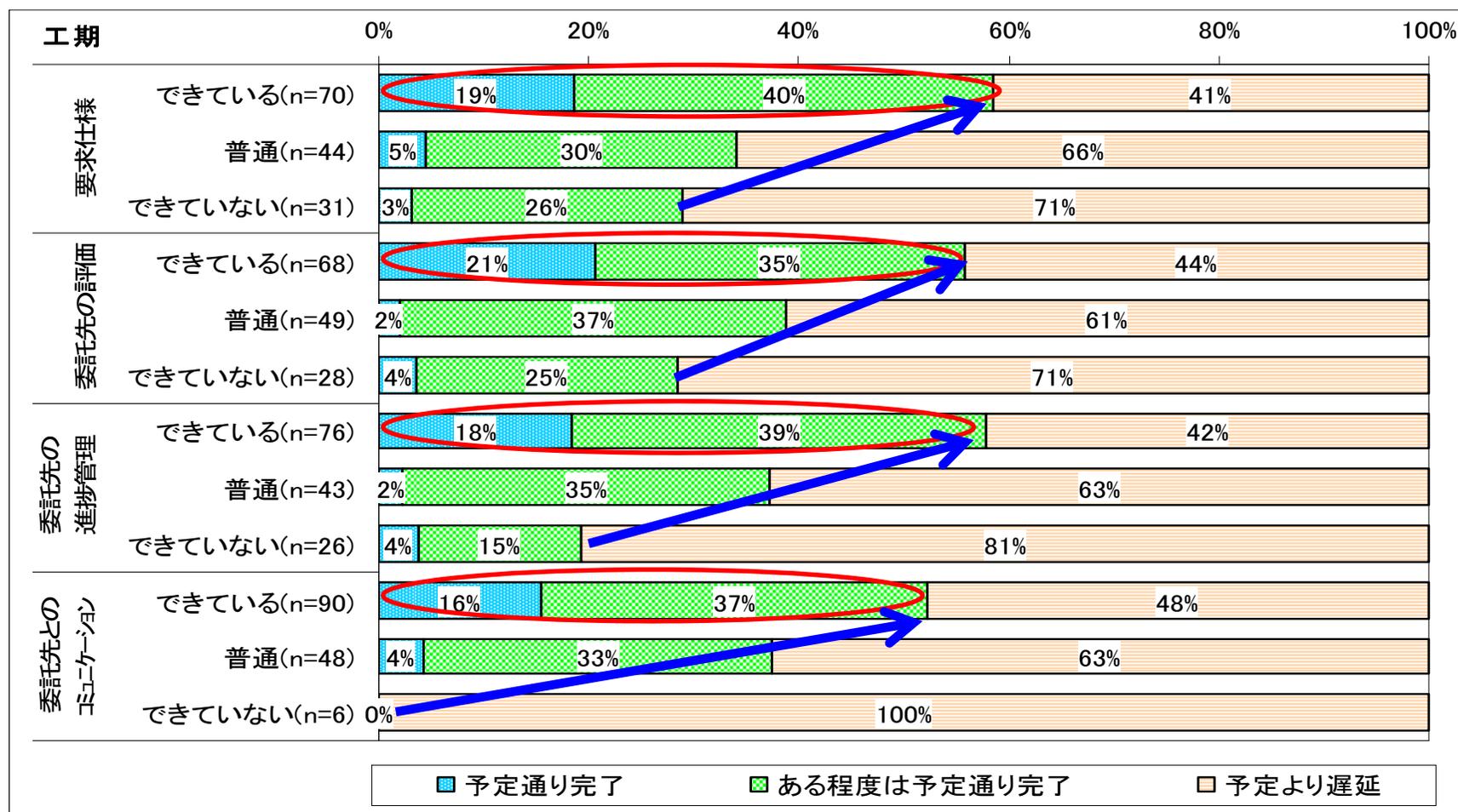
500人月以上の大規模プロジェクトでは、1/3が品質に不満 品質への不満も増加傾向にあるので、更なる改善が必要

システム開発の品質



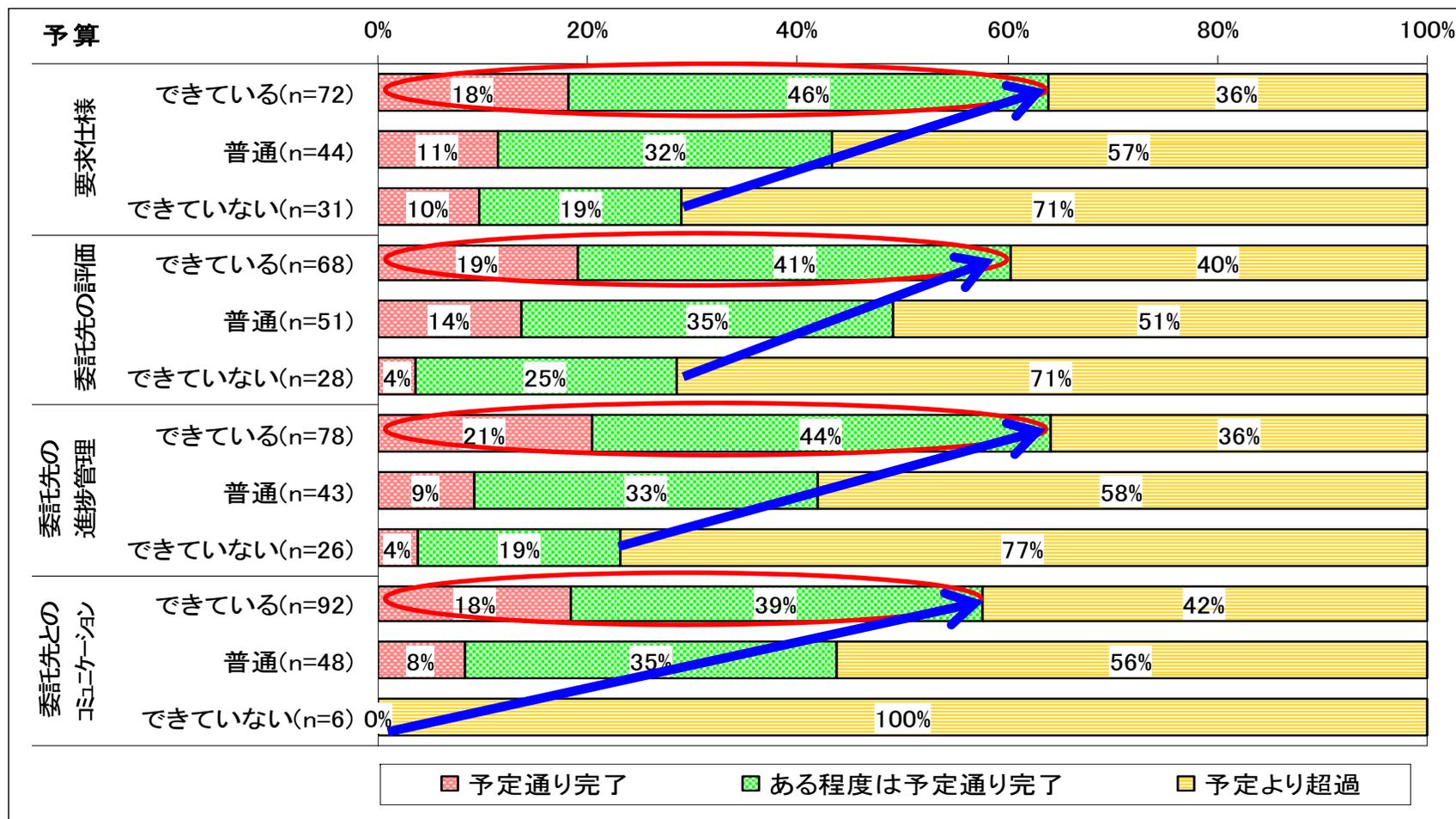
大規模プロジェクトの工期は、 発注者としての対応ができていない企業ほど予定通りに収まる

発注者としての対応と大規模プロジェクト(500人月以上)の工期



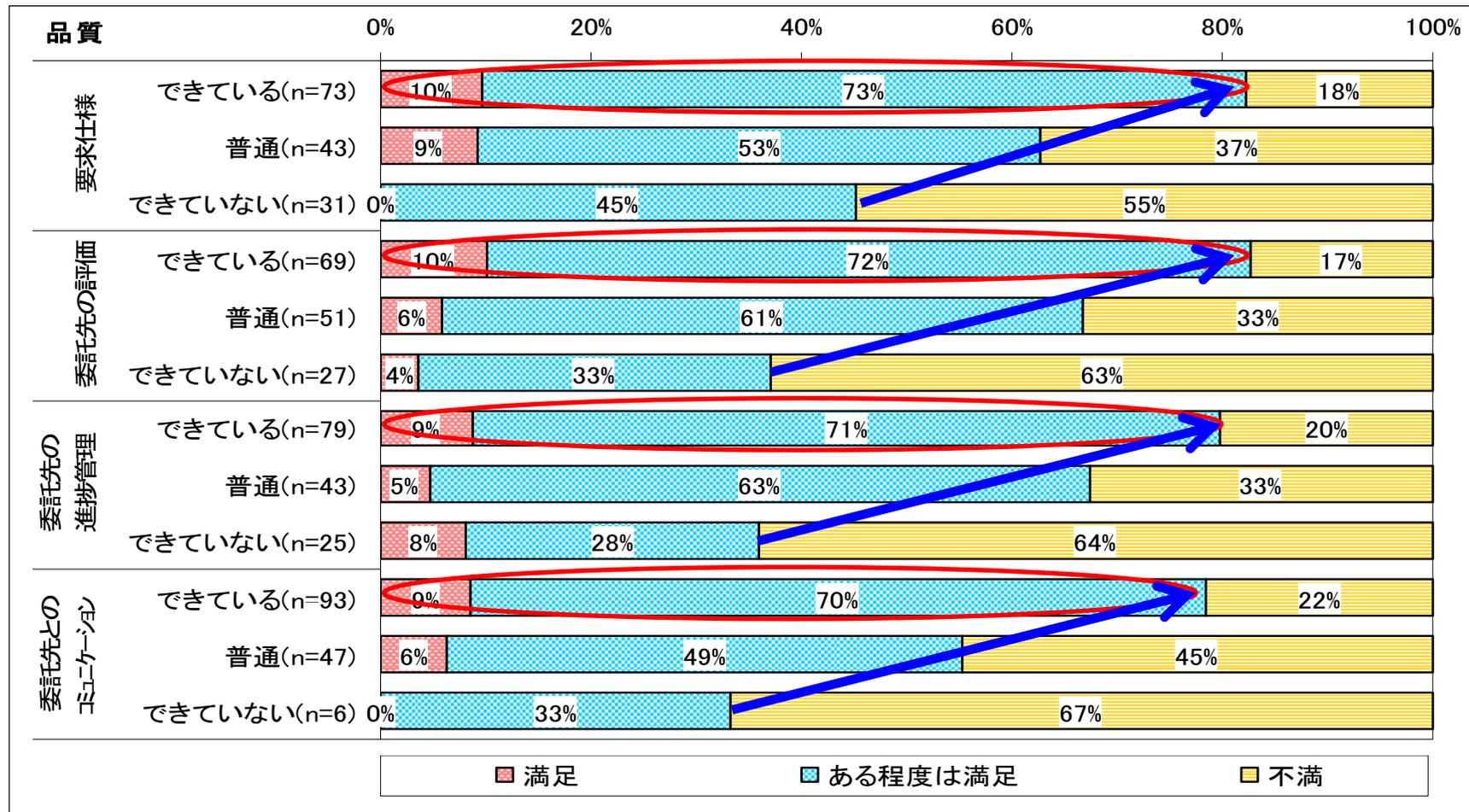
大規模プロジェクトの予算についても工期と同様に、 発注者としての対応ができていない企業ほど予定通りに収まる

発注者としての対応と大規模プロジェクト(500人月以上)の予算



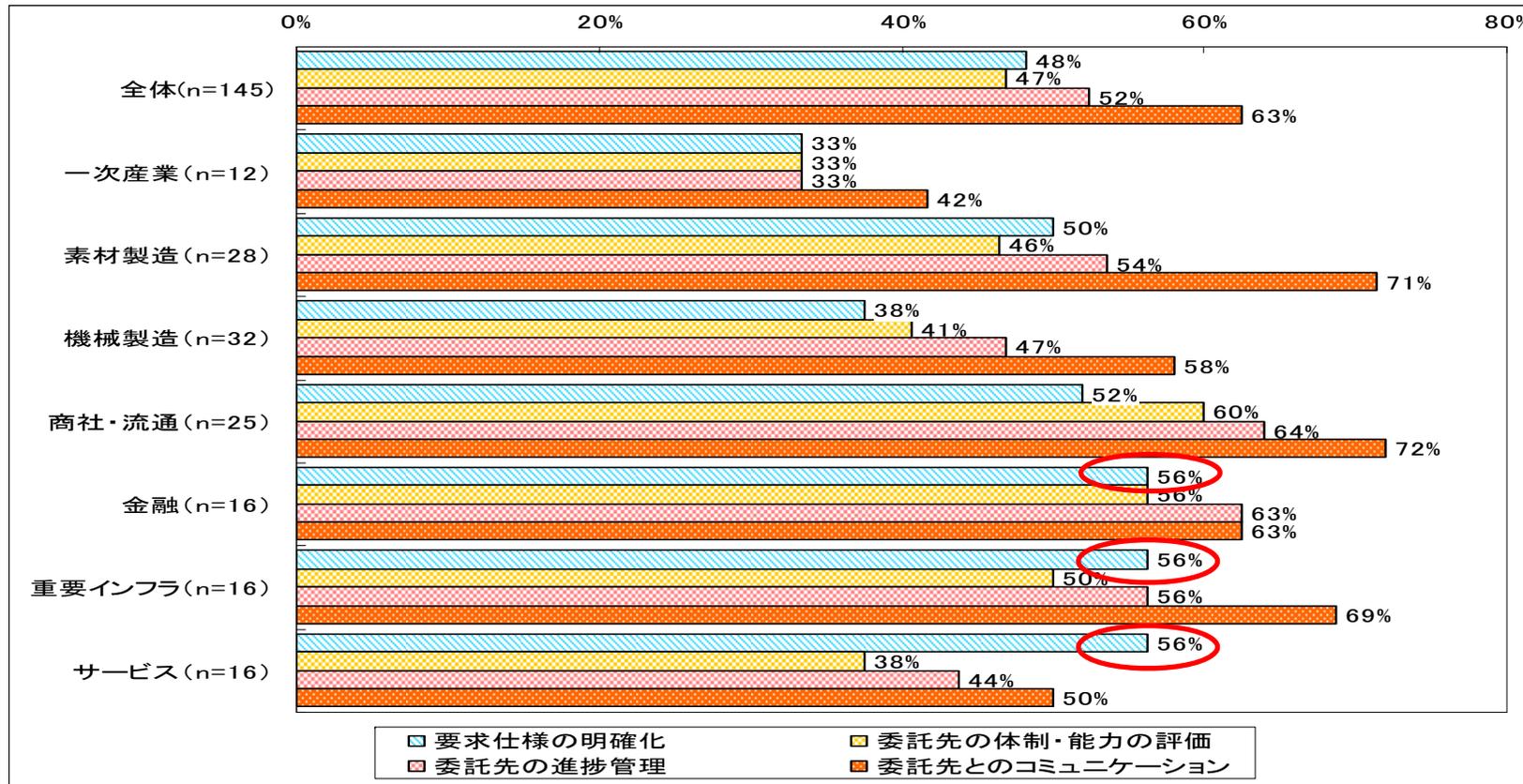
大規模プロジェクトの品質についても工期や予算と同様に、発注者としての対応ができていない企業ほど満足した品質を得ている

発注者としての対応と大規模プロジェクト(500人月以上)の品質



「金融」「重要インフラ」「サービス」は要求仕様の明確化が進んでいるがそれでも6割弱、ユーザーはこの課題に挑戦する必要がある

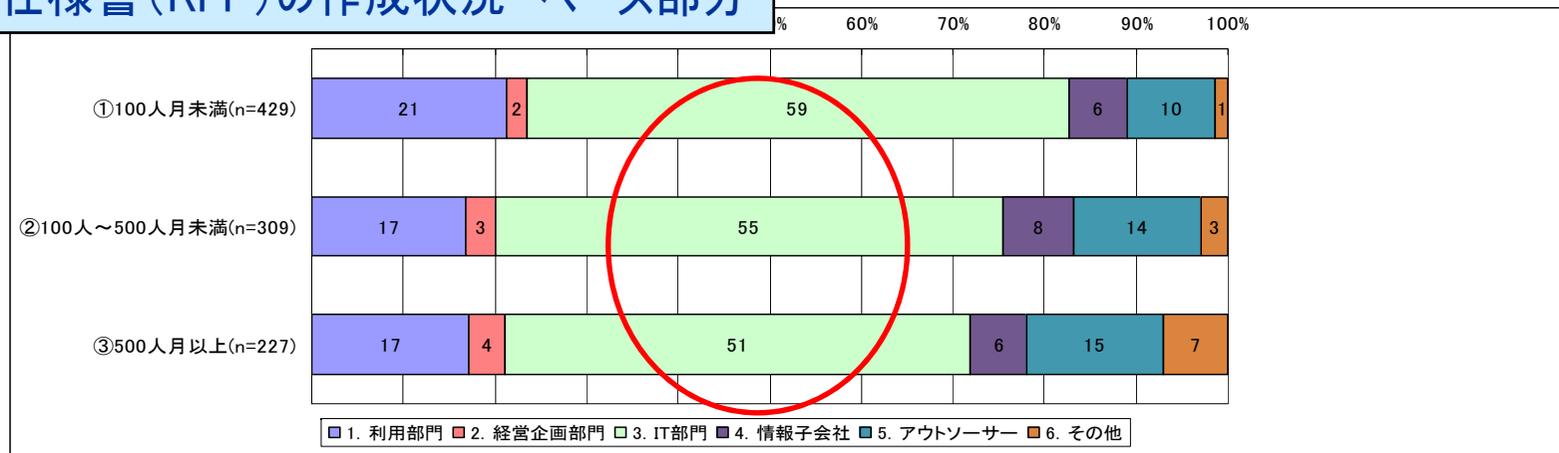
業種別発注者としての対応(対象: 500人月以上の大規模プロジェクトを抱える企業)



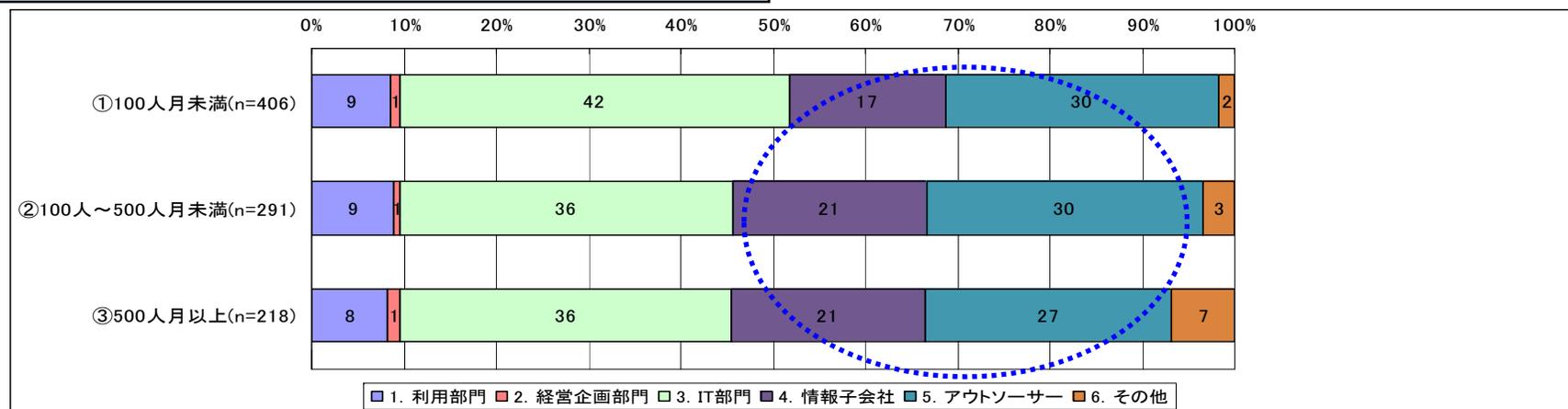
・委託先に対するきめ細やかな対応が大規模プロジェクトを成功に導く要因のひとつである。

要求仕様書(RFP)の「ベース部分」はIT部門が主体で作成 「詳細部分」の作成にはアウトソーサーと情報子会社が加わる

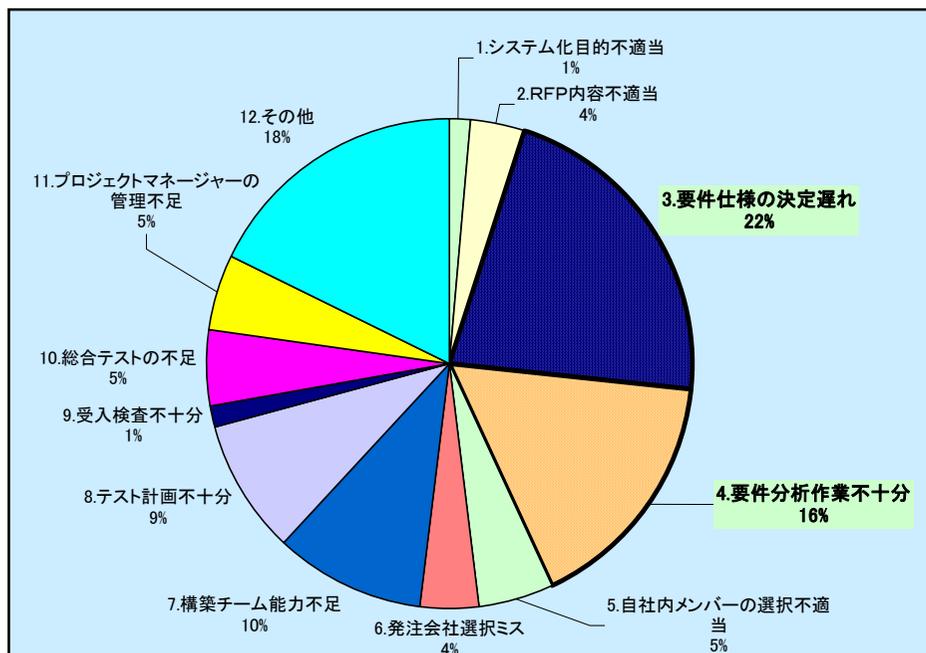
要求仕様書(RFP)の作成状況 ベース部分



要求仕様書(RFP)の作成状況 詳細部分

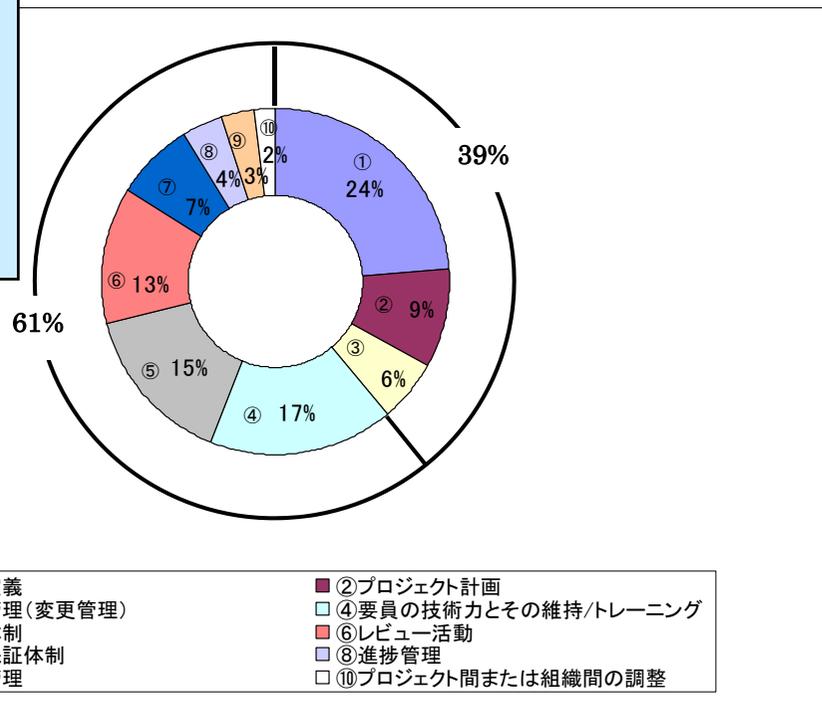


遅延理由分析



要件定義不十分などの
ユーザーの責任は40%

図表 工期遅延理由
(ソフトウェアメトリックス調査)

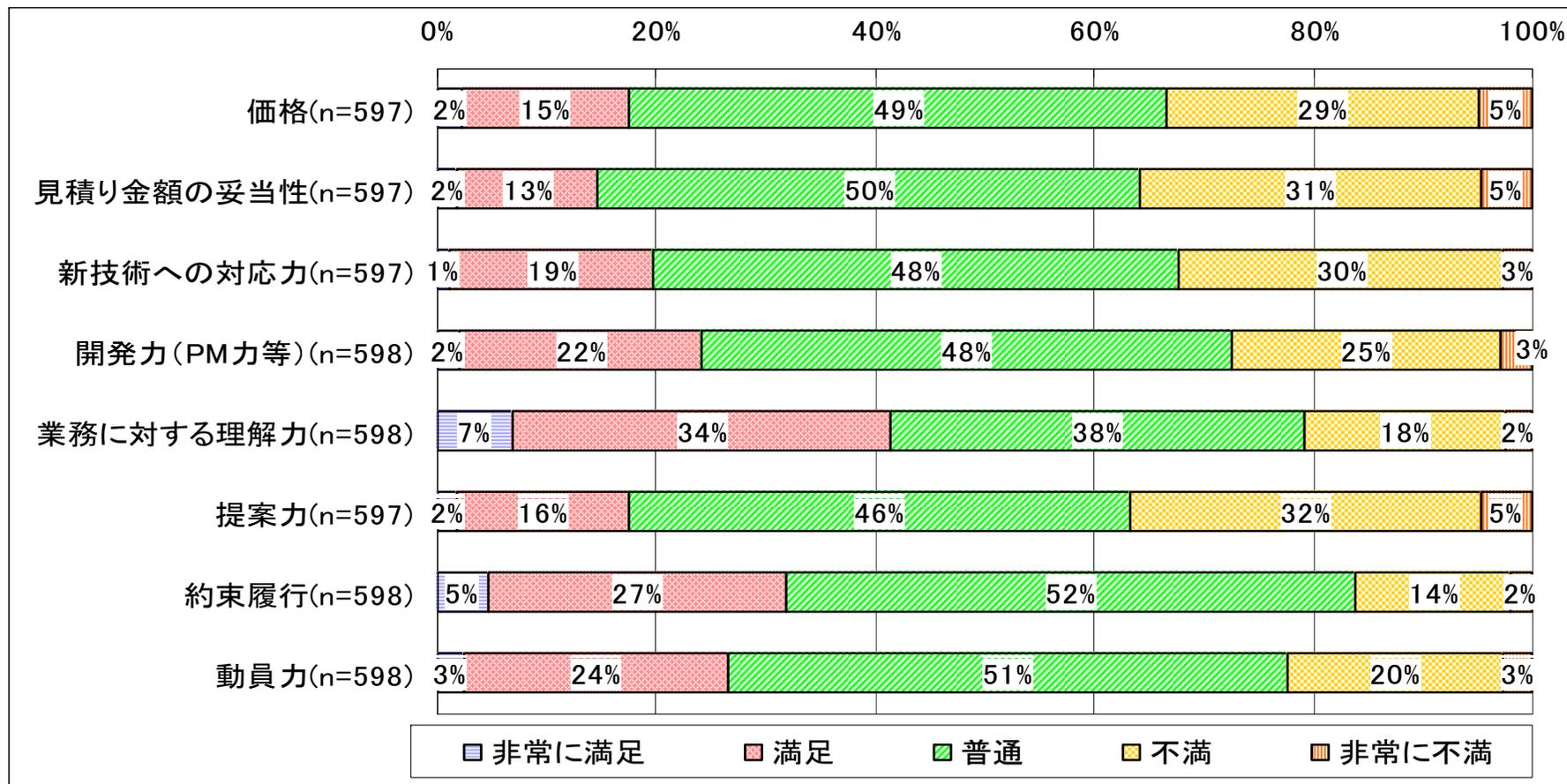


図表 品質とコストの関係(JISAデータ)

開発ベンダーへの満足度 言い訳より対策を

主な委託先1社への満足度

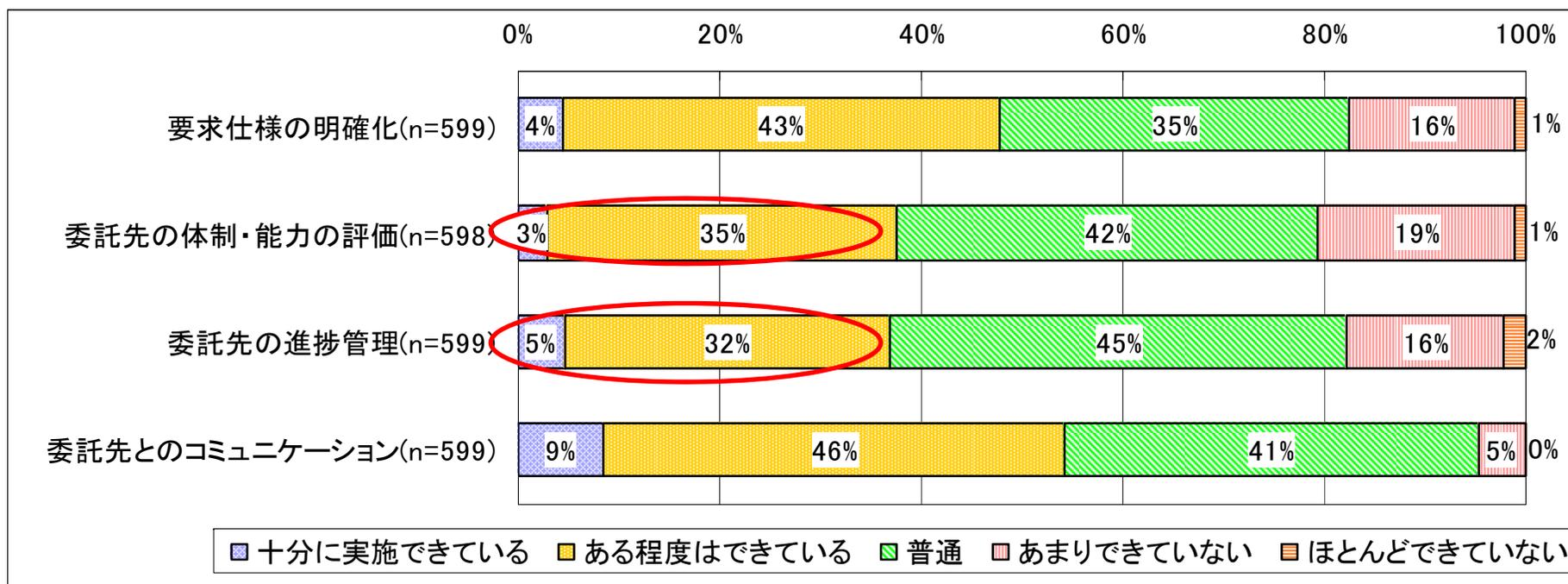
- 提案力不足
- 見積金額の妥当性不明



日本のユーザーは情報産業の高度な技術・品質、安価な高級システムによって支えられている

発注者としての反省点

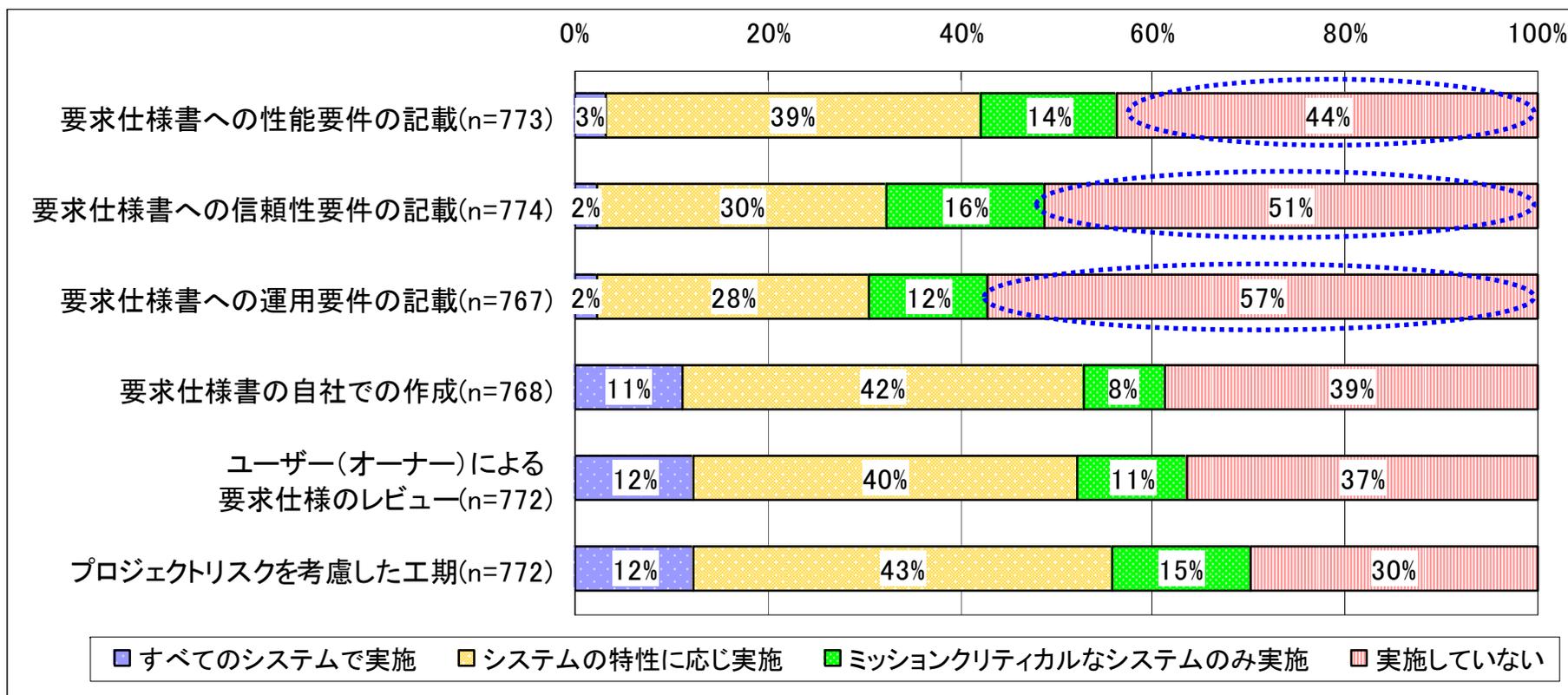
開発を委託する際の発注者としての対応は「委託先の体制・能力の評価」「委託先の進捗管理」が低い結果に



システム仕様の定義を明確にし、適切な役割分担を行う

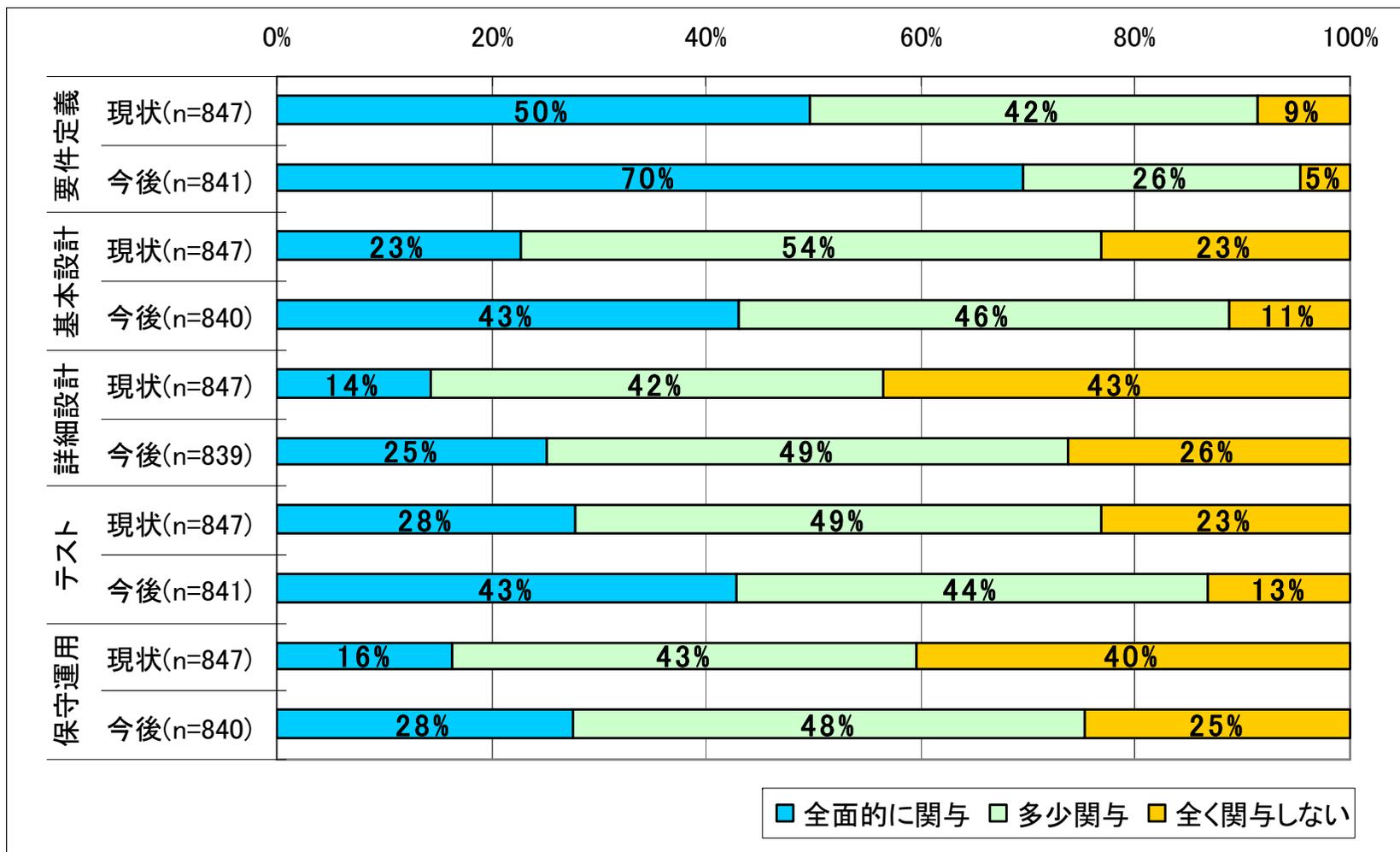
企画・要求仕様段階では要求仕様への 「性能」「信頼性」「運用」の非機能要件の記載充実が課題

企画・要求仕様段階での信頼性向上のための施策



企業IT動向調査2007

ソフトウェア開発における利用部門の関与



JUASIT動向調査2003より

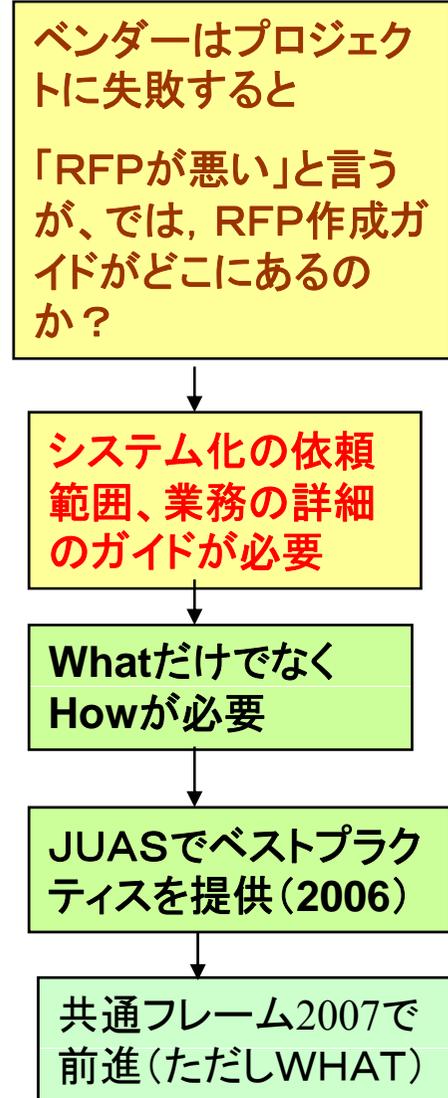
いずれの段階でも今後の方向性は、「全面的に関与」したいという企業が増加

エンドユーザーによるビジネスシステム定義の進め方 機能要求、非機能要求

ユーザーとベンダー間での前提条件の明確化(1)

要求仕様書標作成における役割分担の標準 SLCP98より

基本項目	内容	実行責任部門	
		利用部門	IT部門
① システムの概要	システム化の目標・方針	◎	○
	狙いとする効果	◎	○
	運用対象者	◎	○
	既存システムとの関係	○	◎
② 提案依頼手続き	説明会日程		◎
	対応窓口		◎
	提供する資料	○	◎
	参加資格条件		◎
	提案手続き		◎
	選定方法		◎
③ 依頼事項	システム化の依頼範囲 (業務関係)	◎	○
	(システム関係)	○	◎
	依頼内容、業務の詳細 (業務関係)	◎	○
	(システム関係)	○	◎
	システム構成	○	◎
	納期 (本番時期)	◎	○
	(S/W、H/Wの導入時期、テスト、移行時期)	○	◎
	必要な技術・技術者の資格		◎
	成果物、納入品		◎
	修正プロセスへの参加	○	◎
	共同レビュー	○	◎
	工程計画	○	◎
	開発推進体制	○	◎
	開発手法、開発言語	○	◎
	支援ツール		◎
	ソフトウェア製品の使用		◎
品質・性能条件		◎	
保守条件		◎	
運用条件	○	◎	
④ 開発体制・開発環境	役割分担	○	◎
	作業場所		◎
	開発機器・使用材料の負担		◎
	貸与物件・資料	○	◎
⑤ 保証要件	システム	○	◎
	品質保証基準		◎
	システムの性能		◎
	セキュリティ		◎
⑥ 契約事項	デモ・テスト計画		◎
	発注形態	○	◎
	検収・支払条件		◎
	瑕疵担保責任		◎
	機密保持		◎
⑦ その他	著作権など		◎
	用語	○	◎
	外部委託契約に関する管理	○	◎
	リスクに対する相互認識	○	◎
	仕様変更・機能追加などの条件	○	◎



ユーザーとベンダー間での前提条件の原則

1. 利用者は自らが必要とする機能が何であることを開発者に要求する。
ただし、ITに詳しい開発SEが明確に定義した方が良い項目はそちらに任せる
 (例:「画面から入力されるデータには新規、訂正、追加、削除がある」
 これらの処理は複雑で難しいので開発者に任せる。「どのようにするか」まで記述することは利用者にとって難しい作業である)
2. 入力データをもとに例外作業が発生する場合は、その例外作業項目を構造図などで必ず記述する。「文章の裏を読め」と言うてはいけない。
3. 例外作業を具体的にシステムにどのような手順にしてプログラムにして取り入れるのかは開発者に任せてよいがユーザーは仕様の確認をすること。
4. ルール化できない例外作業は「異常作業」としてシステム外の扱いとなる。誤解が生じないように極力具体的な表現方法を取る。
5. ベンダーは、ユーザーの条件の記述の曖昧さを指摘し両者で確認すること
6. 非機能要件は具体的な項目に絞り、評価項目を含めて両者の合意をとること
 (JUASのUVC2を参照のこと)

業務システム仕様書の記述レベル

5段階にわたる業務担当部門による業務システム仕様書記述レベル

仕様の責任と記述項目			レベル-1	レベル-2	レベル-3	レベル-4	レベル-5
			ビジネス機能提示	ビジネスプロセス提示	業務フロー提示	個別業務処理提示	個別業務処理/データ項目提示
責任分担	1	ユーザ側責任	RFPLレベルでベンダが提案書をベースに要求仕様書を明確にしシステム構築	既存システム再構築ではビジネスプロセス定義と既存システム仕様提示で発注	As-Isベース機能拡張で業務フローを通常業務/例外業務処理につき提示	To-Beベースの業務処理方法の提示による効果的な発注	To-Beベースでのシステム構築の発注によるシステム構築効率追及
			要求仕様承認の責任	基本設計承認の責任	詳細設計/機能性能の承認	納品仕様/成果物の仕様確認	システム設計の効率化の推進
	2	ベンダ側責任	RFPのビジネス機能からTo-Beのあるべき姿のシステム機能を提案	提示ビジネスプロセスの改革を実現するシステム構築を実現	更なるTo-Be機能への展開とシステム設計と納品物のQCD追求	To-BeベースのIT処理方法からの改善改革と構造的で明快なシステム構築	To-Beベースのシステム機能強化と最適化の追求
			納入/契約仕様の実現	設計、納入品の瑕疵責任	システム設計開発の合理化/コスト低減	綺麗な構造設計/コスト低減	システム設計開発の合理化/コスト低減
A	ビジネス機能関連図				IS部門で企業/事業全体機能定義	IS部門で企業/事業全体機能定義	IS部門で企業/事業全体機能定義
B	ビジネス連携図				業務と対外系/他部門間との連携	業務と対外系/他部門間との連携	業務と対外系/他部門間との連携
C	ビジネスルール定義書				企業/業務上の戦略ルール	企業/業務上の戦略ルール	企業/業務上の戦略ルール
D	システム化目標定義書		業務システム化の目標設定	業務システム化の目標設定	業務システム化の目標設定	業務システムのITベネフィット定義	業務システムのITベネフィット定義
1	ビジネス機能構成表		ビジネス機能の大分類定義	ビジネス機能の中小分類定義	ビジネス機能の細分類定義	ビジネス機能の細分類定義	ビジネス機能の細分類定義
2	ビジネスプロセス関連図			ビジネスプロセス間の関連定義	ビジネスプロセス間の関連定義	ビジネスプロセス間の関連定義	ビジネスプロセス間の関連定義
3	業務流れ図				業務処理フロー指示(含む例外処理)	業務処理フロー指示(含む例外処理)	業務処理フロー指示(含む例外処理)
4	機能情報関連図					DFD方式での上位DFDとして作成	DFD方式での上位DFDとして作成
5	業務ルール定義書					業務処理上の社内ルールを定義	業務処理上の社内ルールを定義
6	個別業務処理定義書					各個別の業務処理手順を定義	各個別の業務処理手順を定義
7	画面/帳票一覧				基本的に必要な画面/帳票一覧	基本的に必要な画面/帳票一覧	基本的に必要な画面/帳票一覧
8	画面/帳票レイアウト				画面/帳票レイアウトを定義	画面/帳票レイアウトを定義	画面/帳票レイアウトを定義
9	データ項目定義書						データ項目の属性を定義
10	運用・操作要件書				業務システムの運用・操作の条件設定	業務システムの運用・操作の条件設定	業務システムの運用・操作の条件設定

図表 業務担当部門による業務システム仕様書の記述レベル(ビジネスシステム定義研究2004)

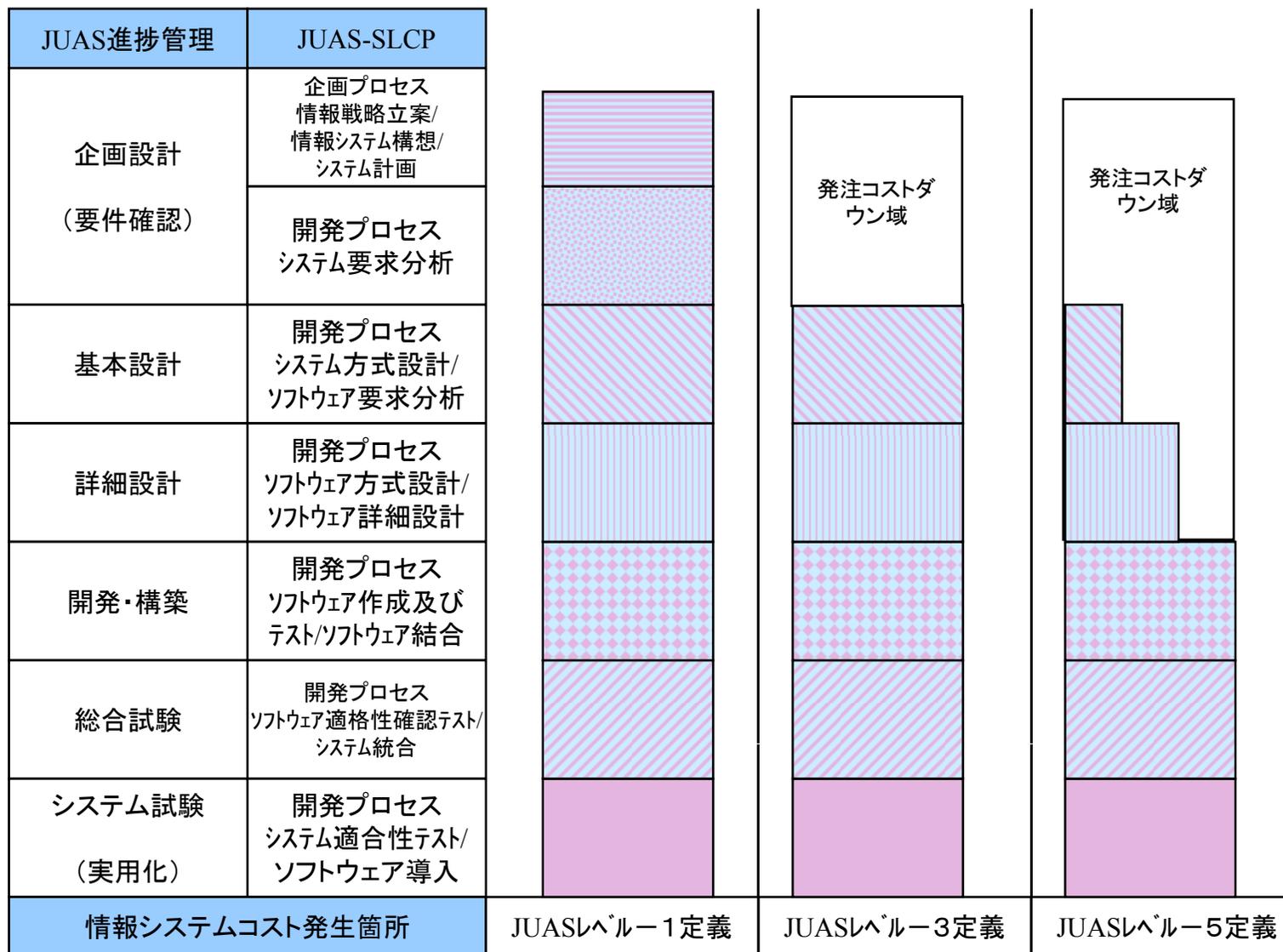
業務ルール定義書

分類	準拠ベース	通例ルール処理基準	例外ルール処理基準	システム化要求		
共通						
	1	顧客管理	1 取引先企業／担当者識別コード			
	2	製品商品サービス管理	1 製品／商品／サービス識別コード			
商流	1	代理店経由販売				
	1	1	営業区分	1 営業テリトリ判別 担当営業選定		
		2	見積作成	1	見積作成依頼受信 見積作成依頼発行	
				2	提案内容 (標準／オプション商品)	1 別途設計し製造する商品
				3	見積単価選定 (顧客別商品別単価選定)	1 特別単価の設定 2 必要コストアイテム見積
		4	納期提案 (標準納期提案)	1 別途設計・製作品納期		
	3	特例契約条件	1 標準契約交渉 (標準/オプション商品)(標準QCD対応)	1 事業部決済条件の交渉 2 企業のリスクで交渉	人間系で対応	
	4	与信審査依頼	1 顧客与信限度 (規定による与信限度)	1 将来ビジネス勘案で裁量		
	5	受注伝票	1 受注契約 (標準QCDの受注)	1 別途設計・製作のQCD	受注センタで対応	
	6	納期回答	1 納入納期確約 (標準納期で受注)	1 部材、生産枠の不足調整	受注センタで対応	
7	納品検収条件	1 標準品質の工場出荷/検収	1 納品先納入後テスト検収			
8	請求条件	1 標準品の月間納品物の請求 (月次請求書処理)	1 特注品の進行基準の請求			

図表 業務ルール定義書(事例:製造業—営業関連業務)(ビジネスシステム定義研究2004)

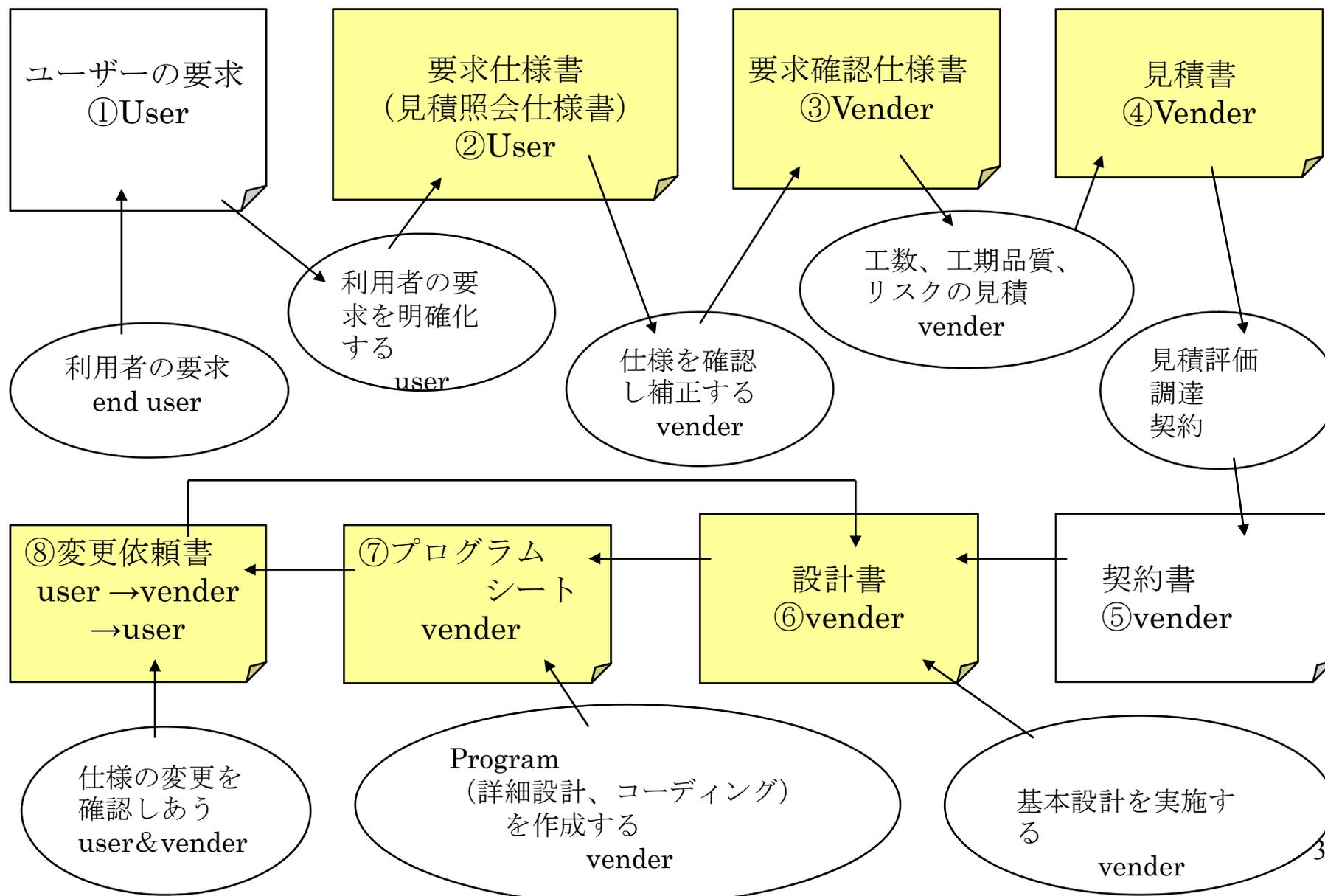
情報システム構築のコストダウンの追及

ユーザーは発注コスト低下、高品質、短工期、使いやすさなどの効果を知る



図表 JUAS業務システム見積り照会によるコスト効果(ビジネスシステム定義研究2004)

UVC要求仕様の明確化とトレース方法



文章問題(エレベーター)

例題4:エレベータ(要求と要求仕様)

問題1:駅の11人乗り、最大積載量700キロのエレベーターに乗ると、最後の一人が「重量オーバーです」とアナウンスされて、降ろされることが、頻繁に発生する。そのために誰が降りるのか?乗車した人の葛藤が始まり、時間をロスすることが多い。これを防ぐ要求仕様を「清水様式」に則り作成してください。

問題2:この様式についての是非と記入上の注意事項を整理してください

ヒント:加重計が600キロを越えたら「あと一人で満員です。次の回までお待ちください」と乗車された方にお伝えする。

- ・乗った人は全員合格であり、ドアはしまるので、降りる人はいない。したがって時間もかからない。誰が降りるべきか?乗客間で葛藤が始まることはない。
- ・疑問 最後に乗った人が、曙のような大型力士だったらどうなるか?

文章問題(エレベーター) カテゴリー 積載制限

管理項目	要求番号	区分	要求機能
□□□□	SK01	要求機能	乗り込んだ客の重量を予測し次ぎの一人が乗ったら満員になると予測された場合は警告を発する
		理由	お客が乗り込んでから、降りてもらう不快感と時間のロスを避ける
□□□□	SK01-1	要求機能仕様	お客が乗り込んできたら、その都度時間をカウントし始めると同時に体重を積算し、(制限重量-最後の一人分の余裕)以上になった場合は「これが最後の方です」とアナウンスして伝える
		理由	乗ってから降りてもらう手間を省くため
□□□□	SK01-2	要求機能仕様	前の客が乗り込んでから3秒以上たっても次の客が乗車してこなければ、制限重量以下であることを確認し、ドアを閉め、始動する
		理由	お客が少ない場合は、早めに発車するため

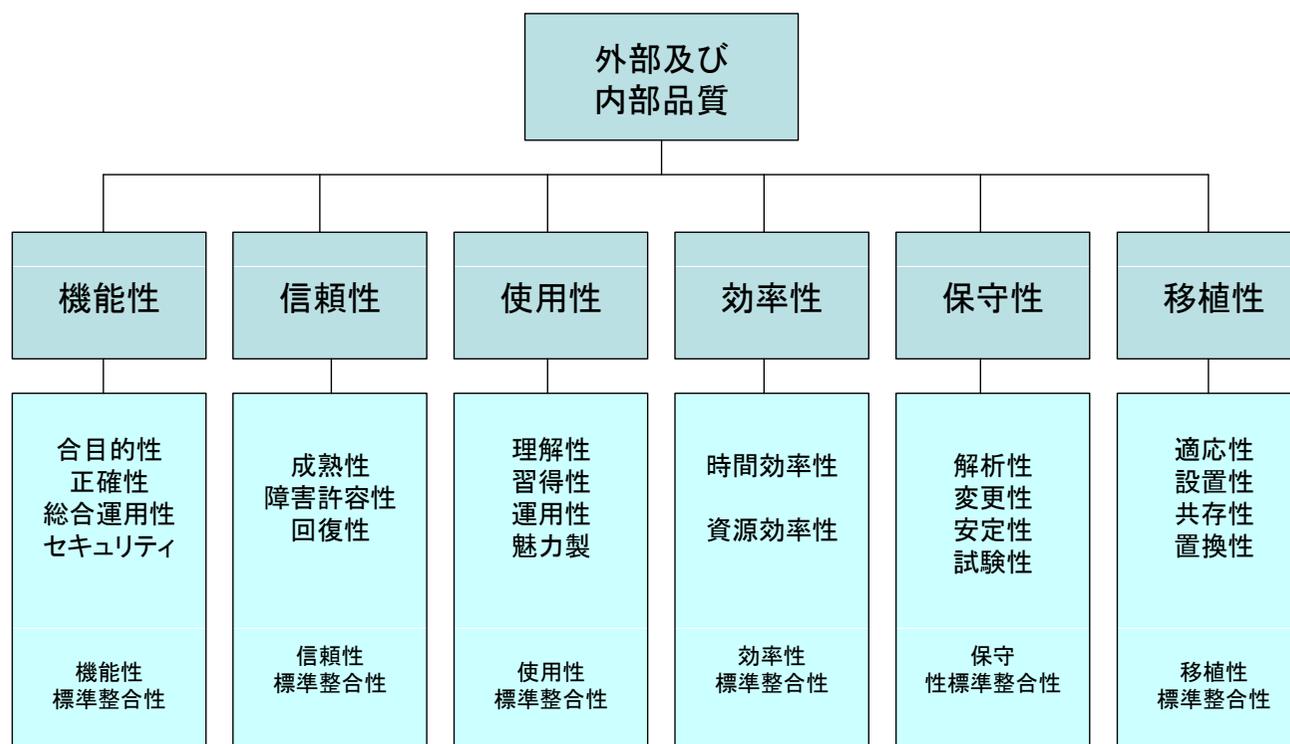
- ・仕様変更率 = 変更仕様数 / 総仕様数 これを一定の率に収める努力をする(管理項目を活用)
- ・要求番号を, RFP、設計書、プログラムシート、変更管理にまで一貫して活用する
- ・(参考)要求を仕様化する技術、表現する技術 清水吉男著 技術評論者
- ・(参考)UVC1 要求仕様書定義ガイドライン報告書 JUAS2007.3発行 (IEEE830STD 準拠)

仕様書の確認の仕方(行間を読む方法)

- ①名詞について：単数か複数か？を確認すること。
例：「出来た製品を倉庫に入れる」この文章を見て、
 - ・倉庫は一つか、複数か？
 - ・製品は一つ一つ入れるのか？ロットにまとめるのか？などの確認をせねばならない
 - ・抽象名刺には細く説明が必要 例：XX管理、管理システム、xxxなど
- ②代名詞の使用はできるだけ避ける 「この、その、あの」「これ、それ、あれ」
- ③形容詞、副詞について あいまいな形容詞、副詞は、すべて確認する 例：十分な在庫
- ④助詞の使い方も注意 「ので」「から」、「は」と「が」、「に」「へ」の使い方
- ⑤その他の表現注意事項
 - ・出来るだけ能動態で書き、受動態を避ける。主語、述語が明確な方が誤解されない
 - ・単文、重文、複文の使い分け（単文優先）、冗長性の排除、〇〇を□□する、
単文・・・パソコンが、壊れた（主語+述語）
重文・・・パソコンはリクエストし、サーバーは答える（主語+述語）+（主語+述語）
複文・・・私が欲しいのは このパソコンです（主語+述語）+（述語）
- ⑥用語を一貫して用いる。異なった人々が作業分担するときは、データ・ディクショナリーが有効
- ⑦名詞間の関係を確認するためにはERD「データベース論理図」が有効
- ⑧データ要素が「どこで作成、参照、削除されるか」の明確化には設計段階のCRUD分析が有効

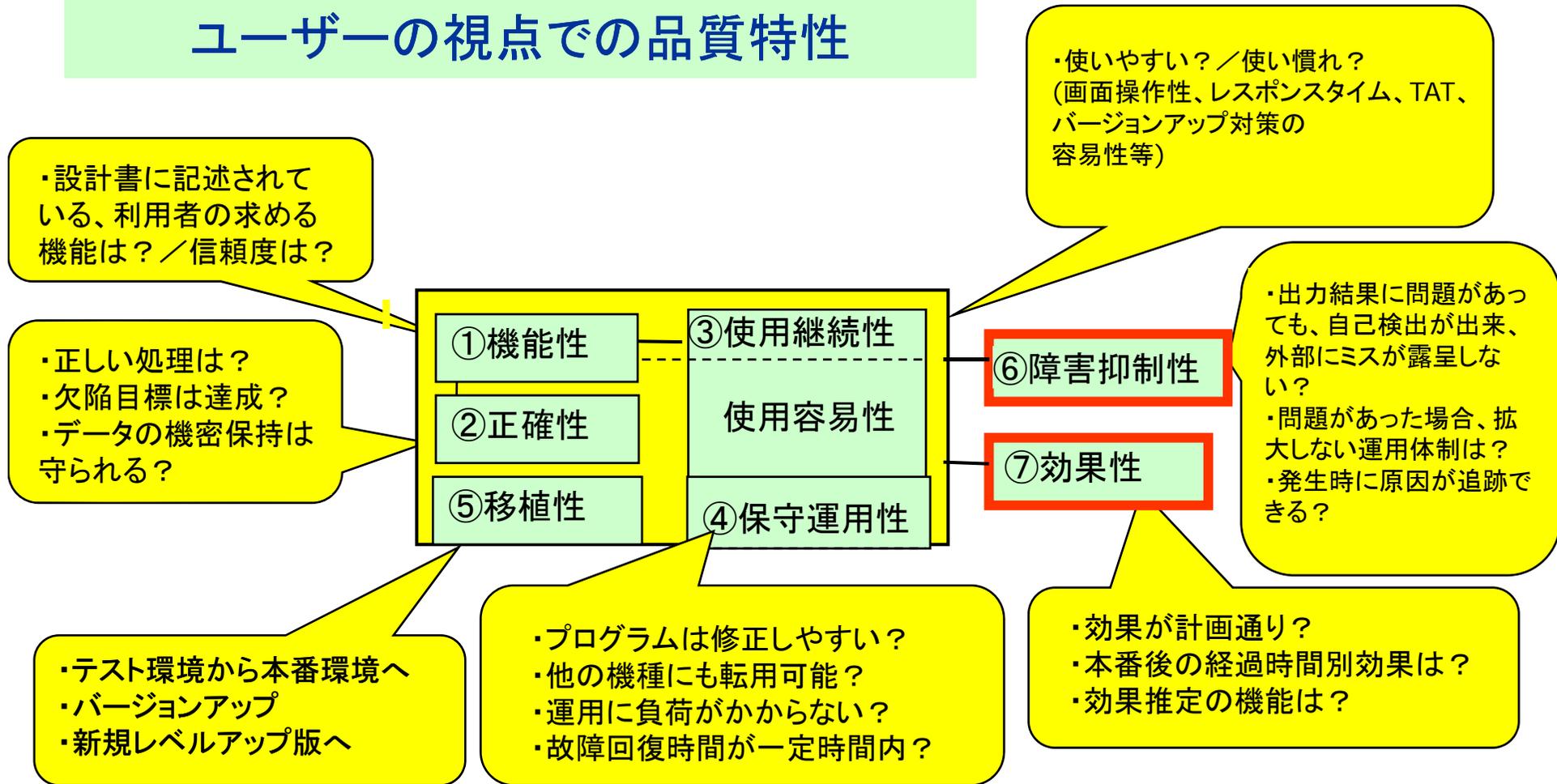
非機能要求とは何か

- ISO/IEC 9126-1:2001 (JIS X 0129-1:2003) によるソフトウェア品質の定義



(「日本工業標準審議会審議、「ソフトウェア製品の品質—第1部 品質モデル JIS X 0129-1:2003 (ISO/IEC 9126-1:2001)」、日本規格協会、平成15年。」より) 40

ユーザーの視点での品質特性



※利用者の立場に立つと、特に上記の6特性が重要。
 ※(JIS X0129-1994)の品質特性は機能性(5)、信頼性(3)、使用性(3)、効率性(2)、保守性(4)、移植性(4)の6特性と21の副特性(括弧内個数)から構成されている。

非機能要件の定義



提案依頼書、要求定義書、設計仕様書、ソースコードなどに適用。
ISO/IEC9126-3

運用を意図したシステム環境でソフトウェア製品を実行するときに測定。
ISO/IEC9126-2

現実のシステム環境で製品が効率性、生産性、安全性、満足性の目標を達成状況を測定する。ISO/IEC9126-4

- ・各特性項目について内部品質、外部品質、使用品質を一貫して整理したものではない。
- ・これら測定法は、すべてを網羅したものとする意図はない。この測定法を修正するか、ここに記載されていない測定法を選択してもよい(ISO/IECTR9126-3内部測定法 序文より)
- ・使用範囲を狭めれば評価項目を整理しやすい(JUASの方式)

UVC2プロジェクト
の特徴



非機能特性
(10種230項目)

- ・全プロセスを通じた、一貫した評価指標とチェックポイントを提供
- ・障害抑制性、効果性、運用性、技術要件などをユーザーの視点から補足
- ・各プロセスの作業内容の解説はまだ一部しかできていない

非機能定義

要求仕様書に書かれるもの (IEE98)
1.機能
2.外部とのインタフェース (人間、ハードウェア、他のソフトウェアなど)
3.Performance (speed, availability, response time, 復旧時間)
4.Attribute (移植性、正確さ、保守容易性、安全性)
5.設計上の制約

SLOD68にはこのような記述は殆どない

UVC1の非機能仕様 各項目にサンプルの提示あり
1.品質(機能性、正確性、使用容易性・継続性、保守運用性、業務品質保証性、障害抑制性、効果性)
2.サービスレベル(サービス提供時間、レスポンスタイム、バッチ処理実行時間、デリバリータイム)
3.安全性(セキュリティ、停止許容時間、有事対策)
4.キャパシティ(データ量、ユーザー数、トランザクション数、アウトプット数、インプット数)
5.コスト(開発規模、開発ツール、システムライフサイクルコスト、システム寿命)
6.使用基盤要件(インフラ基盤、開発標準、保守運用要件、設備(テスト、本番環境))

共通フレーム2007 各項目の内容記述はない
1.ISO/IEC9126 機能性、信頼性、使用性、効率性、保守性、移植性
2.運用、操作要件 システム作業手順、運用形態、障害復旧時間、災害対策、エンドユーザー操作方法
3.移行要件 移行対象業務、移行対象ハードウェア、ソフトウェア、移行手順、移行時期
4.技術要件 システム構成、開発方式(言語など)、開発標準、開発環境
5.付帯作業 環境設定、端末展開作業、エンドユーザー教育、運用支援

非機能要件の種類

	品質特性	品質副特性	項目数
I S O の 規 定	機能性(functionality)	合目的性(2/4)、正確性(2/5)、相互運用性(1/4)、標準適合性(1/1)、セキュリティ(9/21)	15/35
	信頼性(reliability)	成熟性〔6/12〕、障害許容性(3/3)、回復性(0/6) 信頼性標準適合性(0/1)	9/22
	使用性(usability)	理解性(1/6)、習得性(2/7)、操作性(5/7)、使用性標準適合性(1/1)	9/21
	効率性(efficiency)	時間効率性(5/11)、資源効率性((1/8)	6/19
	保守性(maintainability)	解析性(5/8)、変更性(6/9)、安定性(5/7)、試験性(7/7)、保守性標準適合性(1/1)	24/32
	移植性(portability)	環境適正性(5/5)、設置性(2/2)、共存性(1/1) 置換性(2/2)、再利用性(0/1)移植性標準適合性(1/1)、	11/12
I S O 以 外	障害抑制性	発生防止(12/12)、障害拡大防止(9/9)	0/21
	運用性	運用品質目標SLA(2/5)、運用容易性(0/4) 障害対策(1/5)、災害対策(0/2)	3/16
	技術要件	システム実現方式(1/1)、システム構成(2/8)、システム開発方法(3/9)、開発基準(9/11)、開発環境(6/6)	21/35
	効果性	金額換算可能(0/2)、金額換算困難(0/8)、一般的評価(0/7)	0/17
	合計	上段ISO9126に規定してある指標 下段JUAS独自指標との合計	98/230

指標について

特性	副特性	詳細分析ニーズ(測定目的)	No.	測定項目	項目定義と測定方法	測定尺度と導出式
効率性(D)	時間効率性(コンピュータシステム効率)の評価	コンピュータシステムの効率(リアルタイム処理)	D01	レスポンスタイム	ユーザの入力操作から応答までにかかる時間。	秒

解釈	プロセス							ISO	概要	
	組織規制	要求定義	要件定義	設計	プログラム作成	システムテスト	運用テスト			保守・運用
0~少ないほどよい。	①	①	②	③		④	④	⑤	○	

指標について

● 「プロセス」欄の意味

- ① : 要求仕様の決定(ユーザの役割)
- ② : 要求を実現するための仕様提示(ベンダーSE、またはユーザSE)
- ③ : 設計仕様への盛り込み(および確認)
- ④ : 実装結果の確認
- ⑤ : 最終確認

● 「ISO」欄の意味

- : ISO/IEC TR 9126-2 / -3で定義されているもの
- ◎ : 我々独自の定義

{参照} UVC2非機能要求仕様定義ガイドライン(JUAS発行2008.6)

指標について

特性	副特性	詳細分析ニーズ(測定目的)	No.	測定項目	項目定義と測定方法	測定尺度と導出式	解釈	プロセス							ISO	摘要	
								組織規制	要求定義	要件定義	設計	プログラム作成	システムテスト	運用テスト			保守・運用
効率性 (D)	時間効率性(コンピュータシステム効率)	コンピュータシステムの効率(リアルタイム処理)の評価	D01	レスポンスタイム	ユーザの入力操作から応答までにかかる時間。	秒	0~少ないほどよい。	①	①	②	③		④	④	⑤	○	
			D02	スループット	一定の時間内に、何件の処理を終えることができるか。	件数/時間。あるいは件数/秒	0~多いほどよい。	①	①	②				④	⑤	○	
		コンピュータシステムの効率(バッチ処理)の評価	D03	ターンアラウンドタイム	ユーザによる作業開始時(またはジョブの起動)から、求められた情報の出力を終了するまでの時間。	秒、分、あるいは時間	0~少ないほどよい。	①	①	②				④	⑤	○	
			D04	スループット	一定の時間内に、何件の処理を終えることができるか。	件数/時間。あるいは件数/秒	0~多いほどよい。	①	①	②				④	⑤	○	
			D05	実行時間	コンピュータが作業を始めてから、終了するまでにかかる時間。	秒、分、あるいは時間	0~少ないほどよい。	①	①	②	③			④	⑤	○	
	時間効率性(業務効率)	業務効率の評価	D06	スループット	一定時間内に、何件の業務を処理することができるか。	件数/時間の要素(日、時間、分、あるいは秒)	0~多いほどよい。	①	①				④	④	⑤	◎	
		デリバリータイムの評価	D07	提供業務	情報の提供/伝達を目的とする業務の開始から、利用者の手元に情報/実体物が届くまでに、どれだけの時間がかかるか。	時間(日、時間、分、あるいは秒)	0~少ないほどよい。	①	①				④	④	⑤	◎	
		業務処理時間の評価	D08	処理方式把握・準備時間(平均)	利用者が、ターミナルへの移動、端末のセットアップ、マニュアルの取り出し・参照、コード分区の準備など、作業の準備に要する時間。	秒、分	0~少ないほどよい。	①	①				④	④	⑤	◎	
			D09	処理時間	マニュアル/ガイド等を参照しながら、業務処理を行うのに要する時間。処理中のエラー訂正のための時間を含む。	秒、分	0~少ないほどよい。	①	①				④	④	⑤	◎	

①:要求仕様の決定(ユーザの役割) ②:要求を実現するための仕様提示(ベンダーSE、またはユーザSE)

③:設計仕様への盛り込み(および確認) ④:実装結果の確認 ⑤:最終確認

非機能要件の定義 レスポンスタイム① ユーザーの要求(UVC1より)

	具体的目標値
	2秒
2.個別照会	2秒
3.一覧表示(最大20件)	5秒
4.登録処理	3秒
5.複雑条件検索	10秒
6.回線の前提	1Mbps以上
7.PCのスペック	Pentium4,2GHz以上、メモリは1GB。 WindowsXP Professional 台数3000台
8.サーバ	本番環境(別添構成図参照のこと)

非機能要件の定義

レスポンスタイム② 種類と指定

レスポンスタイムの種類	説明	保証
1: 開発マシン環境 端末機は1台	開発会社のテスト環境1台の端末機でのレスポンスタイム	A
2: 開発マシン環境 端末機はn台	開発会社の端末機をつないでのテスト環境でのレスポンスタイム、(n台同時入力して最後の入力の応答時間)	A
3: 本番環境端末機は1台	本番環境ではあるが、他の入力はない場合のレスポンスタイム	A、B
4: 本番環境端末機はn台 (n<99)	本番環境で同種データn個同時入力した場合の最終テストデータのレスポンスタイム	B
5: 本番環境、端末機はn台 (n>99)	本番環境で同種データn個同時入力し、干渉が起きる他のプログラムが背景にながれている場合の、最終入力のテストデータのレスポンスタイムまたは90%以上の入力が制限時間以下	B
6: 本番環境、端末接続数は (n>100/1秒間)	本番環境で同種データn個同時入力した場合の最終テストデータのレスポンスタイム(nはインターネットからの三桁/秒以上の入力件数を想定した場合)ただし同時に干渉が起きる他のプログラムが背景にながれている。最後の入力データに対するレスポンスタイム または90%以上の入力が制限時間以下	C

A: プログラム, DBの作成方法の確認(保証者はプログラム作成企業)

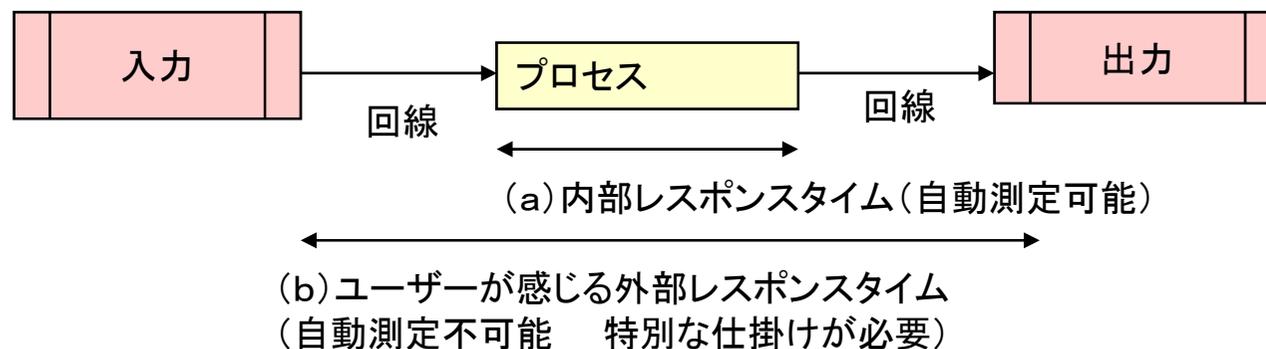
B: プログラム, DBの作成方法の確認(保証者は本番環境提供ベンダー)

C: 本番環境でのレスポンスタイムの確認(対策としてGRIDコンピュータなどの活用も含む)

非機能要件の定義

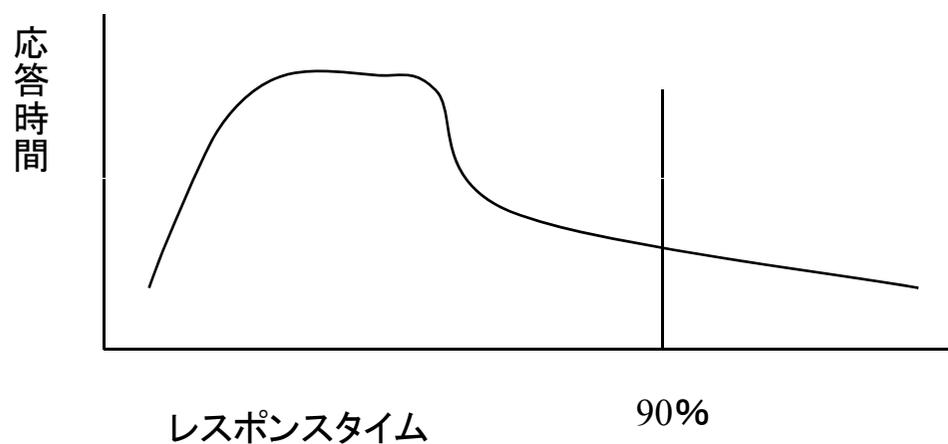
レスポンスタイム③ 種類と指定

①レスポンスタイムの範囲



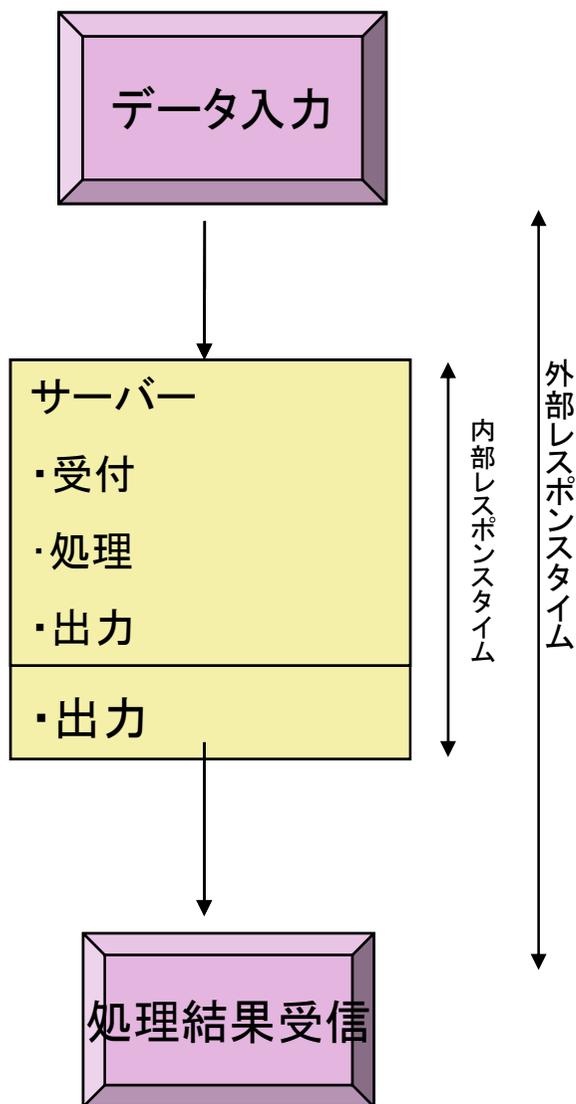
②レスポンスタイムの静的把握、動的把握

- * N個のデータ入力に対して最後の入力の応答時間
- * 通常処理の90%のデータの応答時間



非機能要件の定義

レスポンスタイム④ レスポンスタイムの把握



- ・入力画面表示
- ・データ入力
- ・データ送信
- ・a.出力時間把握

- ・b.受信時間把握
- ・(a-b)時間計算
- ・サーバーに送信
- ・受信結果表示
- ・(a-b) - (サーバー内処理時間)が伝送時間

この機能を、アプリケーションシステムに組み込むことによって、ユーザー体感のレスポンスタイムを把握できる。

しかし一般には行われていない。

(理由)

- ・余分な処理時間がかかる
- ・組み込む操作がわずらわしい
- ・企業内システムでは回線速度は問題にならない

常時このレスポンスタイムを把握しておき、予防保全を実施することが必要。利用者から「最近レスポンスタイムが遅くなった」といわれてから、対策をとるのは遅い。

非機能要件の定義 レスポンスタイム⑤ 稼働率とレスポンスタイムの管理状況

	オンライン稼働率	レスポンスタイム (社内向け)
目標を持っている 企業の割合	74%	13%
実績を把握してい る企業の割合	43%	20%

- ・レスポンスタイムの定義から整理・確立する必要がある
- ・インターネット活用システムのレスポンスタイムはほとんど保証されていない

▪ほとんどのシステムは、End to Endのシステムを把握しておらず、利用者に「遅くなった」と言われて始めて、調査を始める受身型のシステムになっている

非機能要件 例:レスポンスタイム

(高速応答を必要とするプログラムのみ)

フェーズ	詳細機能定義	作業	役割分担	
			USER	Vender
組織規制	オンライン・システムのレスポンス・タイムについて、組織として「原則z秒以内にレスポンスを返す」ことを、組織の方針として明確にしておくのがよい。	①	提示	確認
要求定義	<p>1. 組織としてレスポンス・タイムの方針がない場合、方針があってもそれより早いレスポンスを要求する場合、あるいは「処理が複雑」などの理由で組織の方針としたレスポンスの実現が困難と思われる場合などでは、別途要求定義書上に、この情報システムで必要とするレスポンス・タイムを明記する。実際はプログラムごとに複雑さが異なるので、簡単な問合せプログラムと、データベースをいくつも参照しなければならぬ複雑なプログラムなどで、応答時間を変えざるを得ないことを認識しなければならない。</p> <p>当該システムすべてが高速レスポンスタイムを要求する必要はない。使用環境・頻度や顧客の特性に合わせて、指定のプログラムごとに高速レスポンスを要求すればよい。</p> <p>2. 要求の仕方としては、<u>「1分間のトランザクション数がxxx件までの場合、yy%以上の応答がz秒以内で返ること」という形が良い。</u></p> <p>3. <u>クライアント機で送信完了から受信開始までの時間（付7-2で②で示したもの）、それが困難な場合には、サーバで電文を受け始めてからレスポンスを送り終わるまでの時間（付7-2で①で示したもの）の時間を測定する機能をこの情報システムに取り込む必要がある場合には、その旨明記すること。</u></p> <p>4. 応答時間が長い場合には、途中経過を明記する仕組みの必要の有無を、明記すること。</p>	①	提示	確認

フェーズ	詳細機能定義	作業	役割分担	
			USER	Vendor
要件定義	<p>1. 上記組織規制で明記されたレスポンス・タイム、あるいは要求定義段階の2から4までで指示されたレスポンス・タイムを実現するためのハードウェア（ネットワークを含む）とソフトウェア（ミドルウェアを含む）の構成を明らかにし、同時に検証する仕組みを明記する。</p> <p>2. テストデータの準備とテストの責任範囲を明確にする。</p> <p>3. レスポンス・タイムを検証するためのテスト環境などを明らかにし、併せて範囲に応じた責任者を明確にする。</p>	②	確認	提示
設計	<p>1. 厳しいレスポンス・タイムが要求されている場合には、それを実現するために必要な仕組みをこのソフトウェアに組み込むべく設計し、十分に検証する。</p> <p>2. この設計結果を基に、何をどう調整するとレスポンス・タイムを短縮することができるのかについて、あらかじめ理論武装しておくことが望ましい。</p> <p>3. 1分間のトランザクション数がxxx件（要求仕様書で設定した上限の件数）を超えた場合の処置を明確にし、問題が発生しないようにすること。</p>	③	確認	提示

①要求仕様の決定（ユーザの役割） ②要求仕様の決定（ユーザの役割） ③設計仕最終確認様への盛り込み（および確認） ④実装結果の確認 ⑤最終確認

フェーズ	詳細機能定義	作業	役割分担	
			USER	Vender
プログラム作成	上記設計内容を、プログラムとして実現する。早いレスポンスを要求されている場合にプログラム作成と単体テスト段階で考慮すべき事項があれば、それについて十分に考慮する。単体テストで使用するマシーンと本番環境で使用するマシンの差を意識し、総合テストに入ってから問題が発生しないように配慮すること。			確認
システムテスト	実環境に近い環境を作り、場合によればトランザクションの発生について特別の配慮をして、要求定義段階などで約束したレスポンス・タイムが実現されていることをベンダー内テストで確認する。但し関連するシステムとのリンクまで含んでのテストは難しいので、運用テスト段階に任せる範囲を明確にしておくこと			確認
運用テスト	ユーザが受け入れる確認テストの一環として、要求したレスポンス・タイムが得られていることを確認する。（レスポンス・タイムを確認するための環境やトランザクションの準備は、ユーザ側の責任となる。）	④	準備	立会
保守・運用	<p>1. 実際の確認は、この段階でなされることになる。レスポンス・タイムを自動的に測定する機能を情報システムに取り込んだ場合、その機能を使用して測定することになる。その機能を準備しなかった場合には、ストップ・ウォッチを片手に測定する。</p> <p>2. トランザクションの増加などの環境変化に対応するための方策をあらかじめ検討しておき、安くて容易な各種の対応を準備しておくことが望まれる。</p>	⑤	確認	対応

障害抑制性

特性	副特性	詳細分析ニーズ(測定目的)	No.	測定項目	項目定義と測定方法	測定尺度と導出式	解釈	プロセス							ISO	摘要	
								組織規制	要求定義	要件定義	設計	プログラム作成	システムテスト	運用テスト			保守・運用
障害抑制性 (G)	発生防止	品質と価格の関係の評価	G01	品質評価値	高品質ソフトウェアを開発するに当たり、品質について通常のシステムと区別した目標を設定し、その開発予算を増額してプロジェクトを管理する必要がある。 所定の品質目標を達成するための基準の単価をあらかじめ定めておき、その単価と今回の開発での予算単価の割合。	今回の開発での予算単価/品質目標評価基準値に対する単価	0~1に近いほど良い	①	①			④	④	④	⑤	◎	
		工期と価格の関係の評価	G02	工期評価値	高品質ソフトウェアを開発するに当たり、工期について通常のシステムと区別した目標を設定し、その開発予算を増額してプロジェクトを管理する必要がある。 所定の工期目標を達成するための基準の単価をあらかじめ定めておき、その単価と今回の開発での予算単価の割合。	今回の開発での予算単価/工期目標評価基準値に対する単価	0~1に近いほど良い	①	①			④	④	④	⑤	◎	
		要求仕様書の作成についての評価	G03	要求仕様書書式活用度	要求仕様書の書式を定め、それに基づいてこの文書を作成する必要がある。 「要求には理由を付ける」、「仕様の総数を管理し、仕様変更率を抑える」などが、この書式に伴って実施されるべき事項。	1. 要求仕様書書式を活用している/0. いない	0/1 1が望ましい	①	①	②	③		④	④	⑤	◎	
		イレギュラー処理の実装についての評価	G04	イレギュラー処理実装率	イレギュラー処理(エラー処理)を一次対策、二次対策に分けて明確にし、二次スクリーニングについての実装を明確に記述する必要がある。 二次スクリーニングについて記述する必要がある箇所に対する、実際に二次スクリーニングが定義された箇所の割合。	二次スクリーニングの実装箇所数/二次スクリーニングが必要な処理箇所の数	0~1 1に近いほど良い	①	①		①	④		⑤		◎	

①: 要求仕様の決定(ユーザの役割) ②: 要求を実現するための仕様提示(ベンダーSE、またはユーザSE)

③: 設計仕様への盛り込み(および確認) ④: 実装結果の確認 ⑤: 最終確認

特性	副特性	詳細分析ニーズ(測定目的)	No.	測定項目	項目定義と測定方法	測定尺度と導出式	解釈	プロセス							ISO	摘要		
								組織規制	要求定義	要件定義	設計	プログラム作成	システムテスト	運用テスト			保守・運用	
障害抑制性 (G)	発生防止	サブシステム間のトレース機能の実装についての評価	G05	サブシステム間のトレース機能実装率	障害の原因を早期に特定するために、サブシステム間の正当性をトレースできる仕組みが必要である。このトレース機能が必要とされる箇所に対する、実際に実装された箇所の割合。	実装された箇所の数 / トレース機能を必要とする箇所の数	0~1 1に近いほど良い		①	②	③	④		⑤		◎		
		テスト環境準備状況の評価	G06	テスト環境の準備状況	高品質ソフトウェアを稼働させるためには、365日24時間、本番環境と同じ状況で常時テストできる環境(テストツールなどを含む)を保守S/Eのために提供することが必要である。その準備状況を問う。	2.データベース含めた24時間使用環境あり / 1.限られた資源を随時使用可能 / 0.限られた資源を限られた時間だけ使用可能	2 / 1 / 0 2が望ましい。		①	②				④	⑤	◎		
		テストカバレッジについての評価	G07	テストカバレッジ率	ソフトウェアの品質を確保するために、十分なテストを実施しなければならない。プログラムの全ステートメント数に対する、使用したテストデータで1回以上通過したステートメントの割合。	テストで1回以上通過したステートメント数 / 総ステートメント数	0~1 1に近いほど良い	①	①			④	⑤	⑤		◎	100%でも完璧とは言えない。	
		バックアップ機への切替についての評価	G08	バックアップ機への切替熟練度	バックアップ装置への切替、戻しが容易に出来るように、常日頃訓練が実施されていなければならない。その習熟度を問う。	1.常日頃実施しているので問題ない / 0.切替に不安を感じる	0 / 1 1が望ましい	①							④	⑤	◎	組織基準として設定することが必要。
		リスク確認についての評価	G09	リスクの解決率	開発前、及び本番稼働前に、システムのどこに障害が発生する可能性があるのかを関係者間で十分に議論し、リスクを洗い出して対応を明確にし、対策を立てておくことが必要である。指摘されたリスクの総数に対する、対応済みのリスクの数の割合。	対策済みのリスクの数 / 指摘されたリスクの総数	0~1 1に近いほど良い				② ③		① ②	④	⑤	◎		

①:要求仕様の決定(ユーザの役割) ②:要求を実現するための仕様提示(ベンダーSE、またはユーザSE)
③:設計仕様への盛り込み(および確認) ④:実装結果の確認 ⑤:最終確認

特性	副特性	詳細分析ニーズ(測定目的)	No.	測定項目	項目定義と測定方法	測定尺度と導出式	解釈	プロセス							ISO	概要	
								組織規制	要求定義	要件定義	設計	プログラム作成	システムテスト	運用テスト			保守・運用
障害抑制性 (G)	発生防止	テスト確認体制の評価	G10	第三者によるテスト確認体制-1	高品質ソフトウェアでは、第三者のテスト専門機関に委託して、システム障害の可能性についての分析を実施する必要がある。まず、その実施状況を問う。	1.第三者の確認を実施済み/0. 第三者の確認を実施していない	0/1 1が望ましい				③		③	④	⑤	◎	
			G11	第三者によるテスト確認体制-2(指摘事項確認率)	前設問で実施済み(1)の場合に、対策が必要と指摘された箇所の総数に対する、実際に対策を実施した箇所の割合。	対策を実施した数/指摘された総数	0~1 1に近いほど良い				③		③	④	⑤	◎	
		G12	妨害に対する対策の評価	高品質ソフトウェアでは、外部から侵入してくる妨害入力に対する処置を明確にし、対策を立てておかなければならない。防御処置が必要な機能の総数に対する、実際に対策を実施した機能の数の割合。	対策を実施した数/対策が必要な機能の総数	0~1 1に近いほど良い	①	①	②	③	④	④	⑤	◎			
	障害拡大防止策	稼働状況の評価	G13	情報システムの稼働率	稼働すべき総時間に対する、実際に稼働した時間の割合。	稼働した実際の時間数/稼働すべき総時間数	0~1 1に近いほど良い	①			②		④	④	⑤	◎	組織基準として設定することが必要。
		停止防止対策の実施状況の評価	G14	停止時間についての目標達成率	停止時間を極力短くする停止防止の対策実施による効果の把握。総停止回数に占める、停止防止の対策による効果で停止時間の目標内に回復できた回数。	停止時間の目標内に回復できた回数/総停止回数	0~1 1に近いほど良い	①			②		④	④	⑤	◎	組織基準として設定することが必要。
		利用者に迷惑をかけていないことの評価	G15	稼働品質率	利用者に、障害発生による迷惑をかけていないことの確認。総資産規模に対する、障害で利用者に迷惑をかけた回数の割合。	利用者に迷惑をかけた回数/資産規模	0~0 0に近いほど良い	①	①	②	③	④	④	④	⑤	◎	

①:要求仕様の決定(ユーザの役割) ②:要求を実現するための仕様提示(ベンダーSE、またはユーザSE) ③:設計仕様への盛り込み(および確認) ④:実装結果の確認 ⑤:最終確認

特性	副特性	詳細分析ニーズ(測定目的)	No.	測定項目	項目定義と測定方法	測定尺度と導出式	解釈	プロセス							ISO	摘要		
								組織規制	要求定義	要件定義	設計	プログラム作成	システムテスト	運用テスト			保守・運用	
障害抑制性 (G)	障害拡大防止策	稼動初期に起きる故障対策の評価	G16	稼動初期故障対策率	システムが立ち上がらない場合の原因を特定する仕組みを準備していることの確認。 仕組みとしての必要な全機能数に対する、初期状況の確認を実装した機能の数の割合。	初期状況の確認を実装した機能の数／仕組みとしての必要な全機能数	0~1 1に近いほど良い	①	①	②	③		④	④	⑤	◎		
		取り扱い可能なデータ件数把握についての評価	G17	取り扱い可能なデータ件数把握の有無	システムの稼動前に、取り扱い可能な入力データの件数の確認を行う必要がある。 その実施状況を問う。	1. 確認を行っている ／0. 確認を行っていない	0/1 1が望ましい	①	①	②	③	④	④	④	⑤	◎		
		既存システムとの差の確認状況の評価	G18	変更管理確認率	JCL、プログラム、データベースなどについて、既存システムと新システムとの差を確認する必要がある。 変更した箇所の全ての数に対する、確認済みの数の割合。	確認済みの数／変更した全箇所数	0~1 1に近いほど良い	①								⑤	◎	組織基準として設定することが必要。
		ハードウェアのアラームへの対応状況の評価	G19	ハードウェアのアラームに対する対応実行率	ハードウェアの故障警告回数に対して、アクション開始基準と手順を定めて実行する必要がある。 ハードウェアの故障発生総数に対する、基準を守ってアクションを実施した回数の割合。	基準を守ってアクションを実施した回数／ハードウェアの故障発生総数	0~1 1に近いほど良い	①								⑤	◎	組織基準として設定することが必要。
		他社製ソフトウェアの監視状況の評価	G20	他社製ソフトウェア監視状況率	他社製ソフトウェアを使う場合は、現時点でどのようなバグ報告があり、対策がどこまで採られているのかを把握しておく必要がある。 他社製のソフトウェアを利用している総数に対する、確認を実施しているソフトウェアの数の割合。	確認を実施しているソフトウェアの数／他社製のソフトウェアを利用している総数	0~1 1に近いほど良い	①								⑤	◎	組織基準として設定することが必要。

①: 要求仕様の決定(ユーザの役割) ②: 要求を実現するための仕様提示(ベンダーSE、またはユーザSE)
③: 設計仕様への盛り込み(および確認) ④: 実装結果の確認 ⑤: 最終確認

特性	副特性	詳細分析ニーズ(測定目的)	No.	測定項目	項目定義と測定方法	測定尺度と導出式	解釈	プロセス							ISO	概要	
								組織規制	要求定義	要件定義	設計	プログラム作成	システムテスト	運用テスト			保守・運用
障害抑制性(G)	障害拡大防止策	ミスオペレーションの防止策の有効性の評価	G21	ミスオペレーション率	オペレーターへのミスオペレーションに対して、歯止めを利かせる方法を十分に用意しておく必要がある。ここではまず、ミスオペレーションの発生率を把握する。総資産規模に対する、ミスオペレーションの発生回数の割合。	ミスオペレーションの発生回数/総資産規模	0~10に近いほど良い	①							⑤	◎	組織基準として設定することが必要。

①: 要求仕様の決定(ユーザの役割) ②: 要求を実現するための仕様提示(ベンダーSE、またはユーザSE)
 ③: 設計仕様への盛り込み(および確認) ④: 実装結果の確認 ⑤: 最終確認

付10-2 「稼働品質率（G15）」の要求から確認まで

フェーズ	詳細	作業	役割分担	
			U	V
組織規制	高品質であることが要求されている高信頼性情報システムでは、「障害を起こさない」と、「仮に障害が起きても利用者に迷惑をかけない」ことの2つが必要になる。そのような情報システムではこの考え方をベースにして開発と運用を行うということを、組織の方針として明記しておくことが望ましい。	①	提示	確認
要求定義	仮に組織の方針として前記のようなことが明記されていないのであれば、これから開発しようとする情報システムが「高信頼性情報システム」に分類される場合、その方針をこの情報システムの要求仕様書上に明記することが必要である。その上で、組織の方針として明記されている場合も含めて、この指標についての目標値をここで設定する。 既存システムでこの定義がなされていないシステムでも、顧客の要望に基づき高信頼性情報システムのための管理を要請されることがある。その場合はあらかじめ条件を設定し、実現方法を見直すこと。		提示	確認
要件定義	この指標の意味は「資産規模あたりの、運用段階で障害によって利用者に迷惑をかけた回数」である。したがってこの指標の数値を改善するためには、「利用者に迷惑をかけない」ことが必要になる。これには障害を起こさないための仕組みを開発段階に組み込むことに加え、仮に障害が発生してもその影響範囲を局所に限定して、利用者に迷惑をかけないようにする幅広い仕組みが必要である。要件定義の段階でこの両者を実現するための基本方針をベンダーはユーザに提示し、ユーザはそれを了承する必要がある。 分子の「顧客に迷惑をかけた回数」を低減するためには、システム構造、プログラム設計の緻密さ、正確さに加えて、入力データの正しさ、出力結果の正しさ、出力情報に基づきモノが動くことの正当性とその確認方法などの、総ての条件が整わなくてはならない。設計・開発者、計算機室のオペレータ、データ入力者、入出力情報を扱う業務担当者、書類運搬者など幅広い関係者の行動の正当性が保証されて、始めてこの稼働品質率は高いレベルに至る。どのような要因がどのように絡むのか、緻密な分析がこのフェーズで要求される。	②	提示 検討	確認 分析 提案

設計	高信頼性情報システムを開発・運用するに当たって配慮すべき事項を、ベンダーは具体的に設計内容に盛り込み、ユーザはそれを確認する。簡単に停止しないシステム、運用段階でミスをしにくいシステム、運用段階で業務担当者が間違えた場合に自動的に発見できるシステムになっている必要がある。	③	確認	提示
プログラム作成	前記設計に基づいて、ベンダーはプログラムの作成と単体テストなどを実施する。この情報システムが高信頼性情報システムであることに関連し、この作業フェーズで通常の情報システムの開発とは異なる配慮が必要になる箇所がある。	④		確認
システムテスト	作成されたソフトウェアに、いろんな角度からのテストを実施する。この情報システムが高信頼性情報システムであることに関連し、この作業フェーズで通常の情報システムの開発とは異なる配慮が必要になる箇所がある。	④		確認
運用テスト	ユーザは納入を受けたソフトウェアに対して、受入テストを実施する。この情報システムが高信頼性情報システムであることに関連し、この作業フェーズで通常の情報システムの開発とは異なる配慮が必要になる箇所がある。特に運用について、当初の考え方を受け継いで、訓練を含めて十分な準備を行う。	④	確認	立会
保守・運用	準備した事柄を背景にして、情報システムを運用する。この間に障害によってユーザに迷惑をかけた状況を克明に把握し、その回避のための方策を検討するベースとして記録すると同時に、一定時期ごとにこの指標についての目標が達成できたかどうかの評価も行う。	⑤	確認	

- ①要求仕様の決定(ユーザの役割)②要求を実現するための仕様提示(ベンダーSE、またはユーザSE)③設計仕様への盛り込み(および確認)④実装結果の確認
⑤最終確認

リスク管理の2種類

①リスクを見積に繁栄する方法

・見積金額 = コスト + 利益 + リスク

ベンダー責任(ユーザーは文句を言わない。高くて不満なら採用しないだけ)

U/V両者で減少させる

・生産性影響要因 29要因

例:工期 $2.4 \times (\text{投入人月の立方根})$ と比較して

要件定義 設計 製作 テスト

10%以内 +3% +3% +1% +2%

30%以内 +10% +10% +5% +7%

例:顧客窓口特性

期限内に決定する -10%-10% -7%

概ね期限は遅延し決定事項が頻繁に覆る

+20% +10% - +7%

・生産物量影響要因 12要因

注:ジャステックモデルを借用し、JUASで改良

②リスクをトレースしトラブルを防ぐ方法

		H	M	L
システム特性	範囲・規模	10	5	3
	業種・業務	10	5	3
	性能	5	3	2
	基盤	5	3	2
業務特性	責任範囲	10	7	4
	工期管理	10	10	4
	開発体制	10	7	4
	顧客協業	10	5	4
契約特性	請負責任	10	5	3
	契約形態	10	6	3
	契約事項	10	6	3
合計		100	62	35

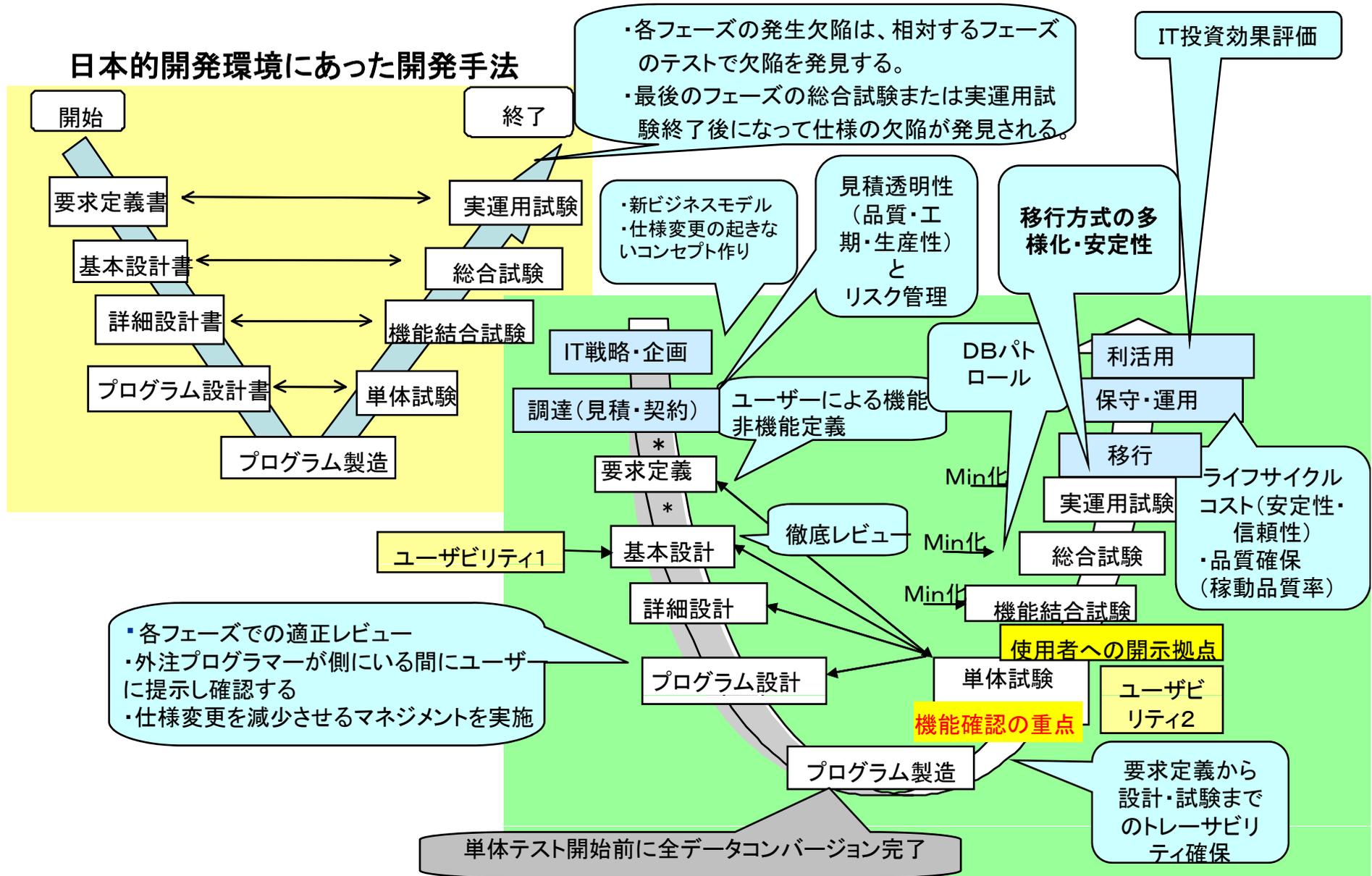
見積、契約、実行計画、基本設計完了、・・・総合テスト完了など開発工程ごとに、このリスク点をトレースし、リスクを確認しながら開発をする方法

システムレファレンスマニュアル第一巻P365

第二巻 p348を参照のこと

Copyright©2008 JUAS, All Rights Reserved

日本の開発環境にあった開発手法

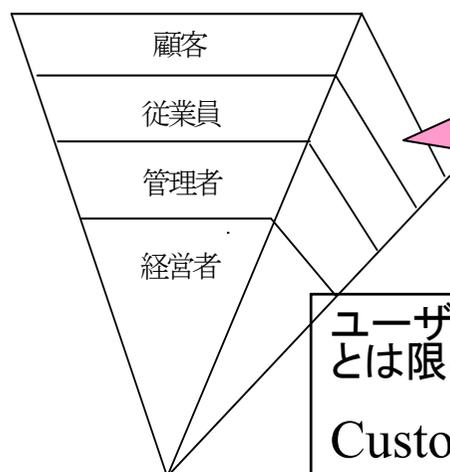


〔参考〕U字型開発手法による改革

- ①企画段階のシステムコンセプト確立・・・
要求仕様の変更は、企画段階のユーザ・リーダーの理解度向上で解消
- ②見積の透明性・・・リスク要因についてのベンダーとの対話
- ③ユーザによるシステム要求・・・機能、非機能仕様の明確化とTraceability確保
- ④徹底レビュー・・・作成時間の10%以上時間をかけて慎重に
- ⑤ユーザビリティの確保・・・ユーザビリティテストによる利用容易性確保
- ⑥UVC判りやすい仕様の書き方と仕様の一貫性
- ⑦目標を持って管理(EASE)・・・目標の有効性とフォロー
- ⑧単体テストの徹底・・・データコンバージョンプログラムは単体テスト開始前に準備
- ⑨単体テスト完了でユーザに開示・・・プログラマーが側にいる間に完了を
- ⑩結合、総合、実運用試験のmin化・・・単体テストの徹底
- ⑪DBパトロール・・・データベース間の整合性チェック
- ⑫移行計画は早期準備・・・開発当初より計画し準備移行方式の多様化・安定性
- ⑬C/O日は定時帰宅を・・・no-trouble で、
- ⑭利活用が最重要・・・投資評価の実施、使いこなしが最重要
- ⑮ライフサイクルコスト・・・利用期間の長期化を配慮した総合コストでの判定
- ⑯稼働率、稼働品質率の確保・・・保守運用段階の目標管理の徹底

逆さまのピラミッド

GMの考え方	視点	サービスマネジメントの考え方
資本、労働生産性	利益コストについての前提	品質の高さ
担当業務実行、職務標準厳守	従業員	真実の瞬間の管理、顧客に直影響
成果	評価基準	顧客満足の数値
標準管理と従業員の服従	管理者	従業員の支持、支援
構造、手続き、規律の管理	組織	従業員の支持と資源配分
組織構造中心経営	経営者	サービス文化の創造と維持



MOT…真実の瞬間はここから始まる。
顧客と接触する第一線の現場従業員を支援するために管理者がおり、その管理者、従業員を支援するために経営者が存在する。

※ MOT=moment of truth

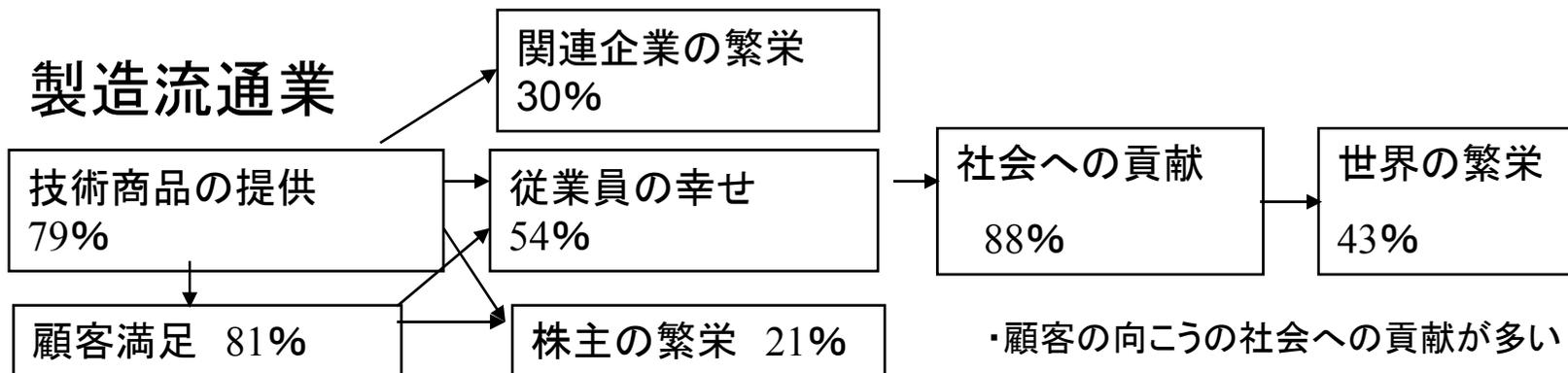
ユーザーは必ず正しい問題認識を持っているとは限らない

Customer is a king and a Partner

①顧客第一主義
お客様のことを第一に考え、顧客要望実現に向けて最大の努力をする

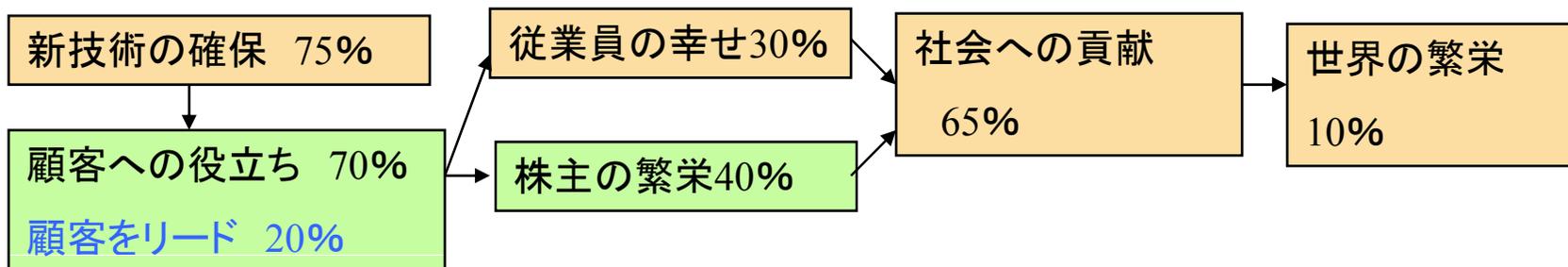
②顧客中心主義
お客さまの要望は素直に受け止めた上で、「顧客が何を望んでいるのか？」を分析し、その要望を実現するための効率的・効果的提案を行い、解決策を実行する。個別の顧客要望に応える解決策を提供する。

企業経営理念2007版

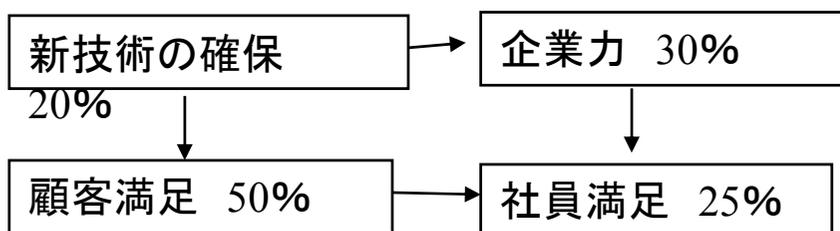


- ・顧客の向こうの社会への貢献が多い
- ・世界市場を視野に入れている

情報産業



情報子会社



情報産業の課題

- ①魅力ある情報産業の創出(従業員の幸せ)
- ②いかに顧客をリードするか？
- ③世界市場に向けての自社の商品の開発

まとめ

①従来の方法が一番良い方法ではない。

もっと楽に、良い品質を生み出す良い方法があると心得えて、努力してください。

②まずは簡単にできるところから、真似てみてください

真似る→工夫する→創造する

守 → 破 → 離

ご清聴ありがとうございました