

JEITAソフトウェア開発モデル契約セミナー

第1部 『トラブル回避のポイント』

ソフトウェア開発モデル契約WG 委員
富士通株式会社 西川 英克

「信頼性の高い情報システムの構築」とは

- ユーザとベンダ間においてトラブルが起こることなく
- ユーザとベンダ間で確定した **仕様** に従い
- 契約に定めた **対価** で
- 契約に定めた **納期** までに
- **品質** の良いシステムを
- 完成させる

よくあるトラブル

「仕様が膨らんだので、契約金額の増額をお願いします。」

「見積と全然ちがうじゃないか！ そんな金払えない！」

「この点の仕様はこうしてくれ。そこの点も。あれも。これも・・・。」

「そんなに仕様変更があると、納期には間に合わないし、契約金額も増額してもらわないと困ります。」

「この作業は順調に進んでますでしょうか？」

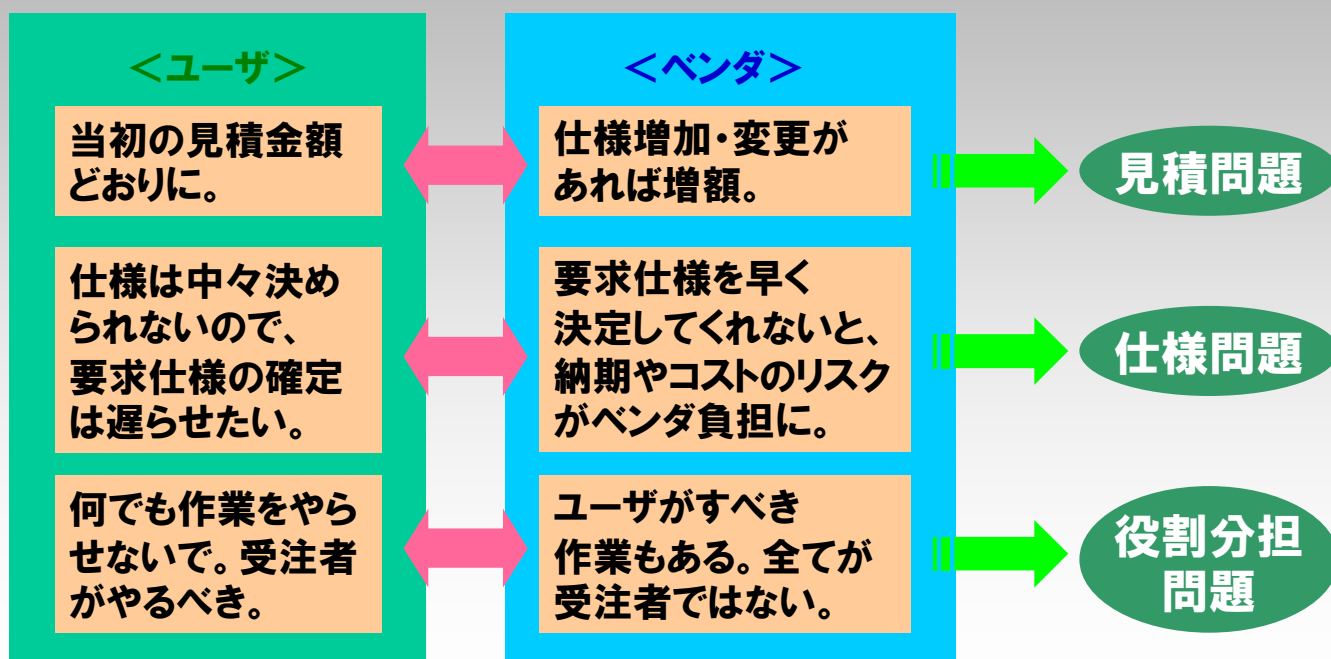
「え！何？こちらが発注してるんだから、当然そちらがやるんでしょ。」

「こんなバグだらけじゃテストにならない。しかも業務も回らないし。」

「バグはすみません。しかし仕様どおりの業務になってるはずですよ。」

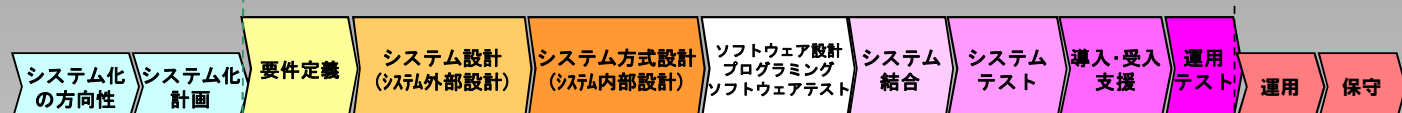
なぜトラブルは起こるのか

ユーザとベンダ間の認識のずれ



見積問題

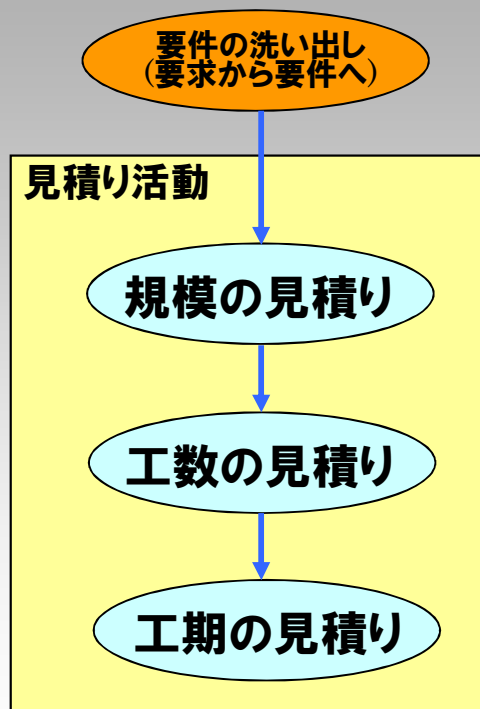
提案段階での見積 ～いわゆる『全体見積』～



本稼働までの開発費だけではなく、運用・保守まで含めた
 ライフサイクル全般の見積が欲しい。

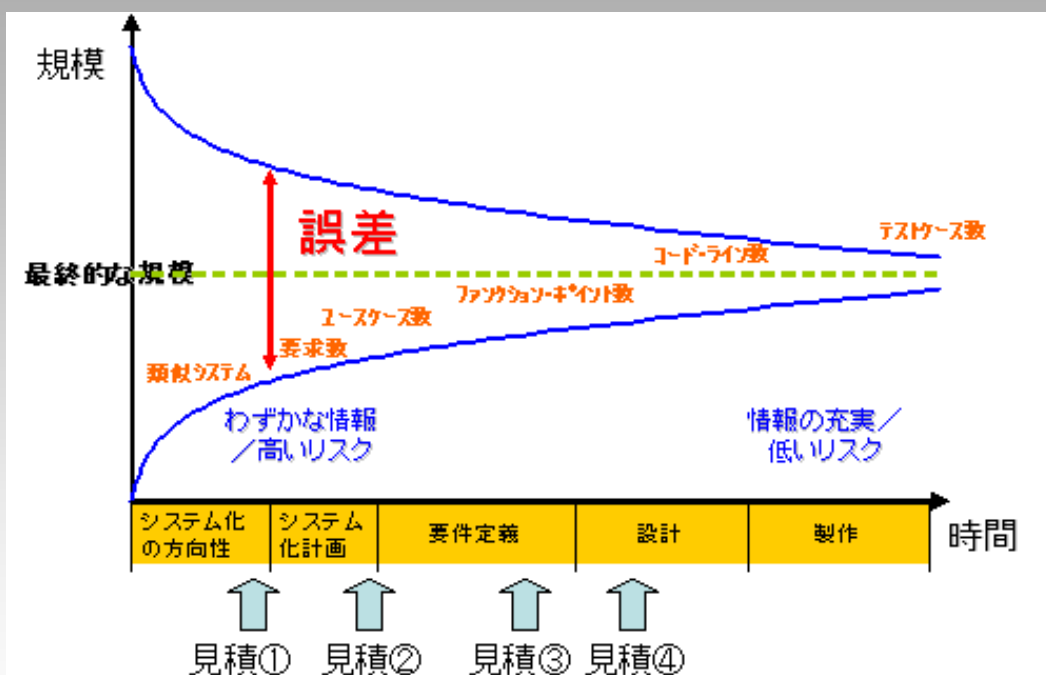
運用・保守まで含めたライフ
 サイクル全般は要らないが、
 本稼働までの開発費の総額
 の見積が欲しい。

ソフトウェア開発の見積りの基本的な手順



出典:「ソフトウェア開発見積りガイドブック～ITユーザとベンダにおける定量的見積りの実現～」
(11頁、独立行政法人情報処理推進機構ソフトウェア・エンジニアリング・センター編)

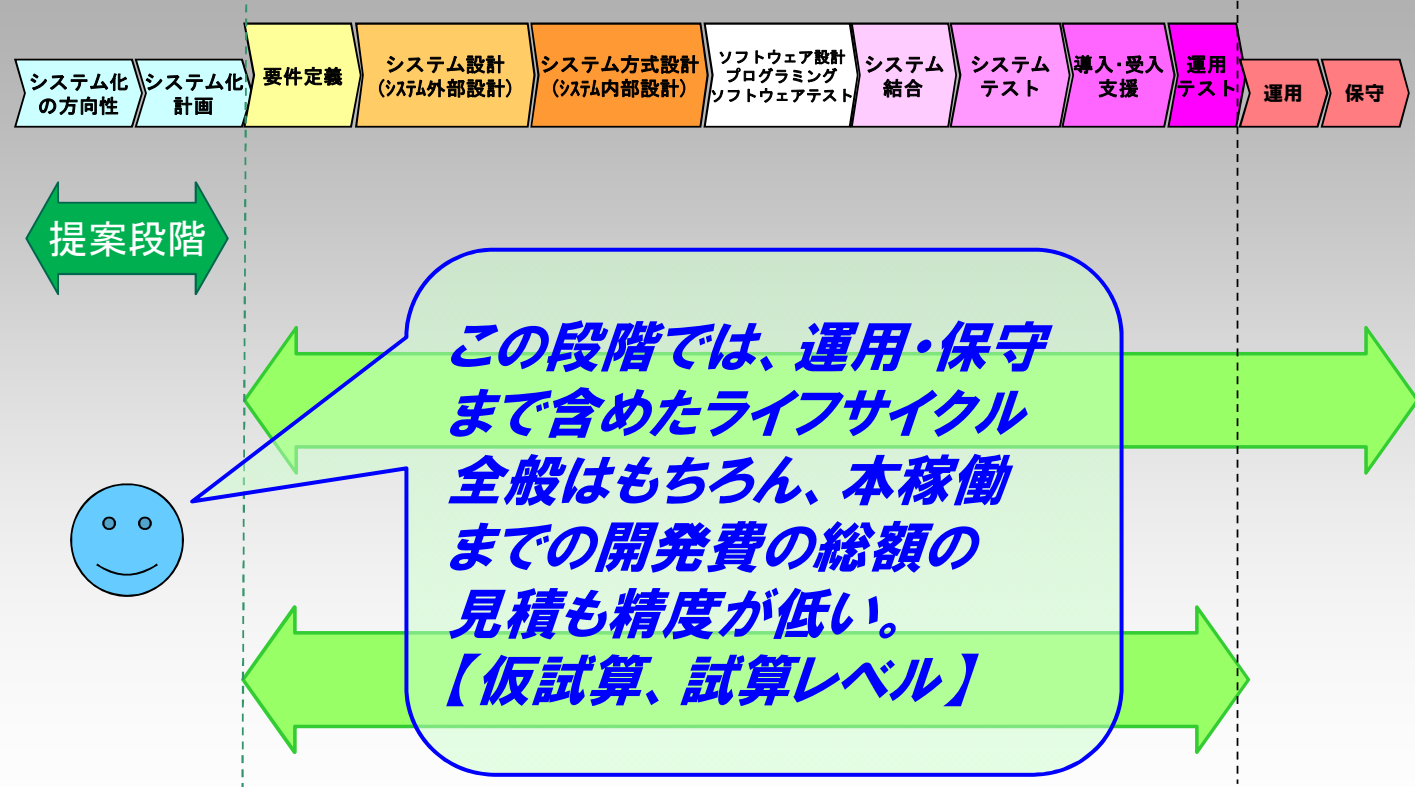
見積り精度



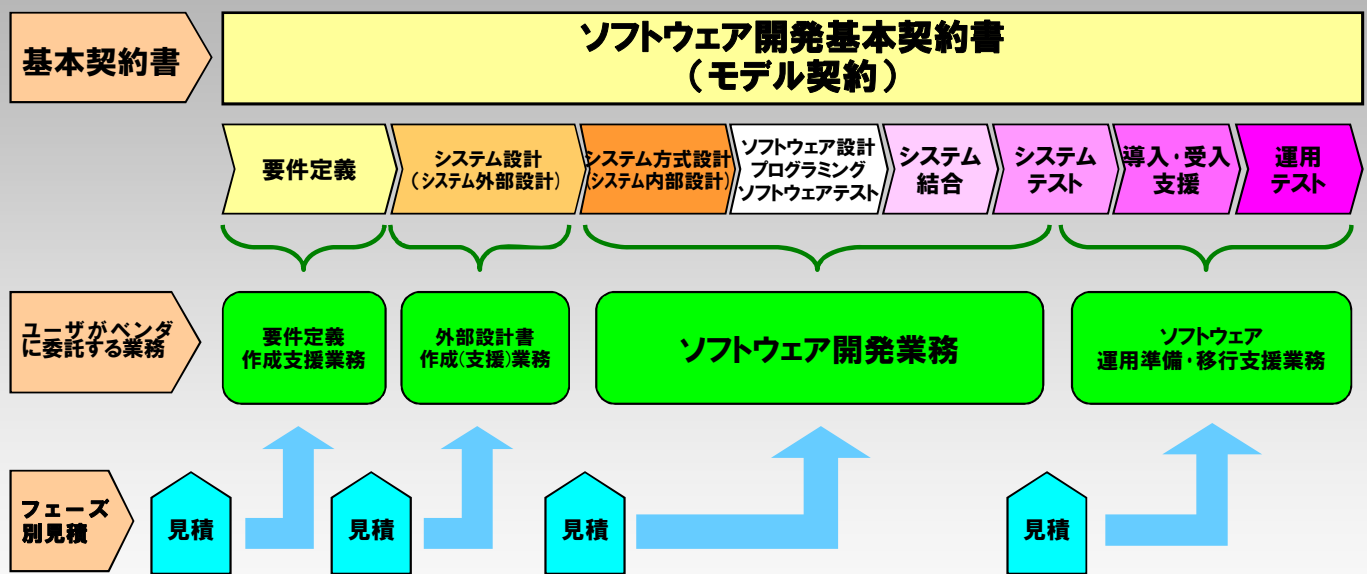
(注: 見積り① = 仮試算見積、見積り② = 試算見積、見積り③ = 概算見積、見積り④ = 確定見積)

出典:「経営者が参画する要求品質の確保～超上流から攻めるIT化の勘どころ～第2版」
(38頁、独立行政法人情報処理推進機構ソフトウェア・エンジニアリング・センター編)

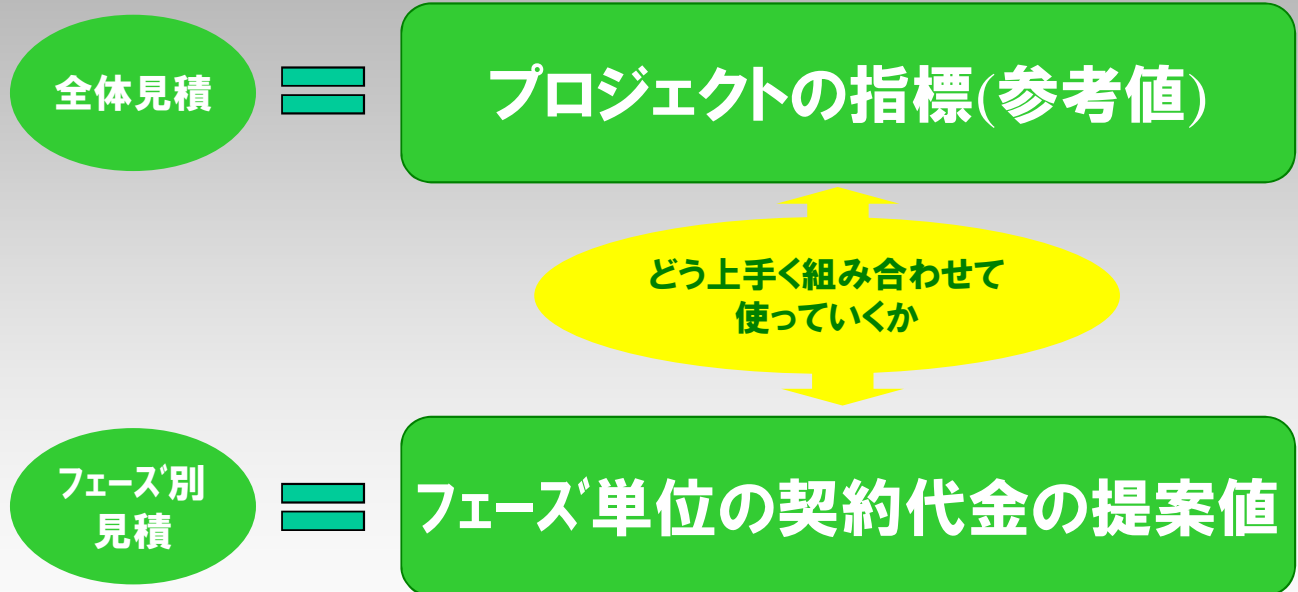
提案段階での見積『全体見積』の精度



フェーズ別見積

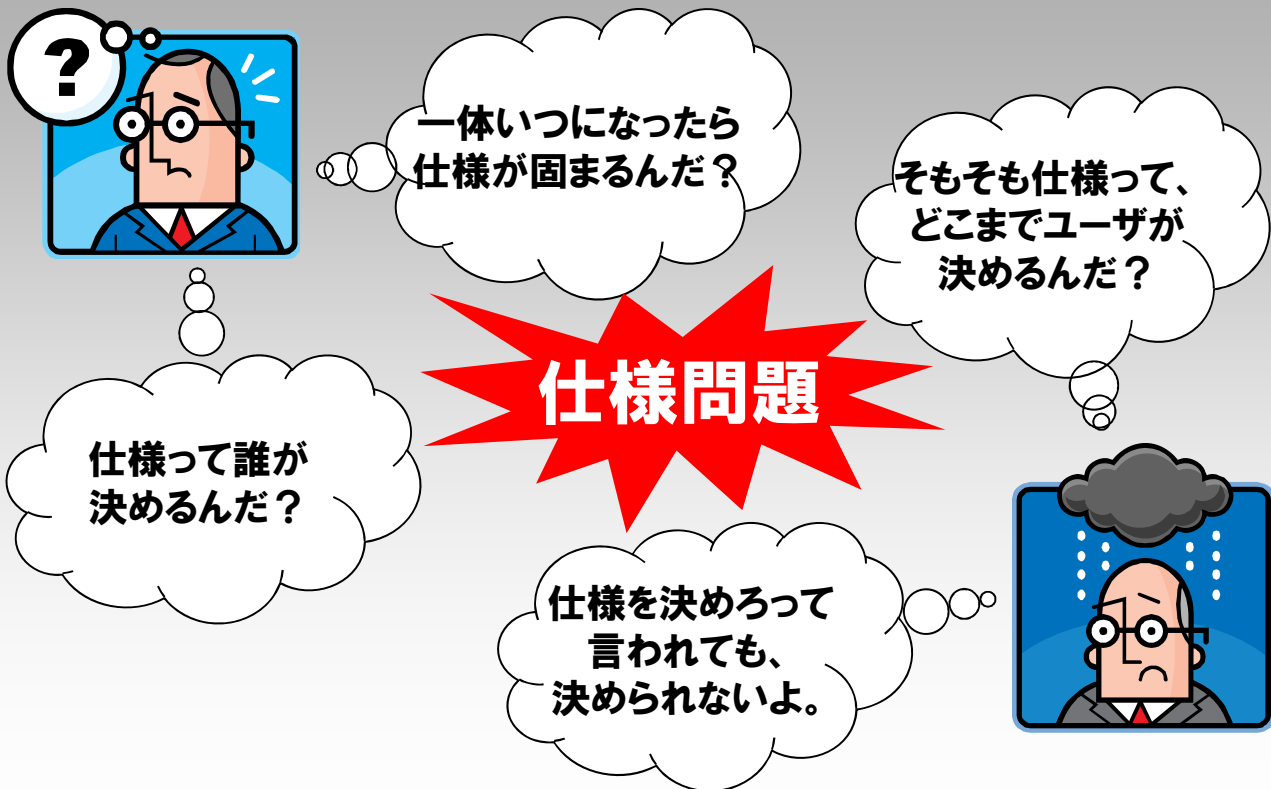


『全体見積』と『フェーズ別見積』



仕様問題

多くのソフト開発トラブルの原因の大きなひとつ



仕様問題とは

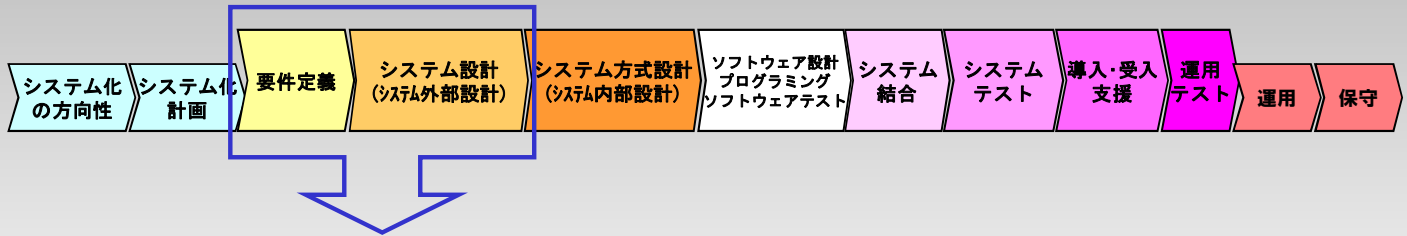
- **仕様の確定**
 - ➡ いかなる「仕様」を誰がいつ確定するのか？
- **仕様変更**
 - ➡ 仕様変更をどう管理していくのか？
 - ➡ 契約条件への影響をどう調整するのか？
- **未確定事項の扱い**
 - ➡ 何が確定していないのか？
 - ➡ 確定した時は条件に影響があるのか？

仕様変更も未確定事項の扱いも『仕様の確定』が大前提

仕様の確定

仕様

開発の目的物として、ユーザの要求がベンダに正しく伝わるようまとめたもの



このアウトプットが仕様

仕様の確定

要件定義あるいは外部設計のアウトプットとして、その後の工程のインプットであるドキュメントについて、ユーザとベンダの間で認識を合わせる。

仕様の確定と多段階契約

要件定義あるいは外部設計における『仕様の確定』は、「何を作るべきか不明確である」というソフトウェア開発トラブルの回避の観点から重要。

- 安易に次の工程に進むことを避け、『仕様の確定』を確実に行う。
- 仕様確定の時点で再度「全体見積」を行い、プロジェクト指標としている提案段階での「全体見積」と比較検討し、「仕様」「費用」「納期」の3点から調整を行う。

「要件定義」、「外部設計」をそれぞれ一つの取引単位とする『多段階契約』

仕様変更と未確定事項

仕様変更

- 確定した仕様のドキュメントがないと、仕様変更かどうか分からない。
- 確定した仕様を前提に、適切な変更管理が必要。

未確定事項

- 「仕様の確定」時に確定できない事項の取り扱いを明確化することが必要。
- 確定した時に契約条件に与える影響を予め見込んでおく。

役割分担問題

役割分担問題の中身

● そもそもソフトウェア開発にはどのような作業があるのか。

➤ 作業体系・内容に関する認識

- ◆ ベンダ各社の標準作業体系・ドキュメント体系
- ◆ IPA SEC「共通フレーム2007」

● ユーザとベンダのそれぞれが行う作業は何か。

➤ 役割分担に関する認識

- ◆ 契約において、作業項目ごとにユーザ・ベンダのどちらが作業主体となるか明示。
- ◆ 工程ごとの作業主体の傾向

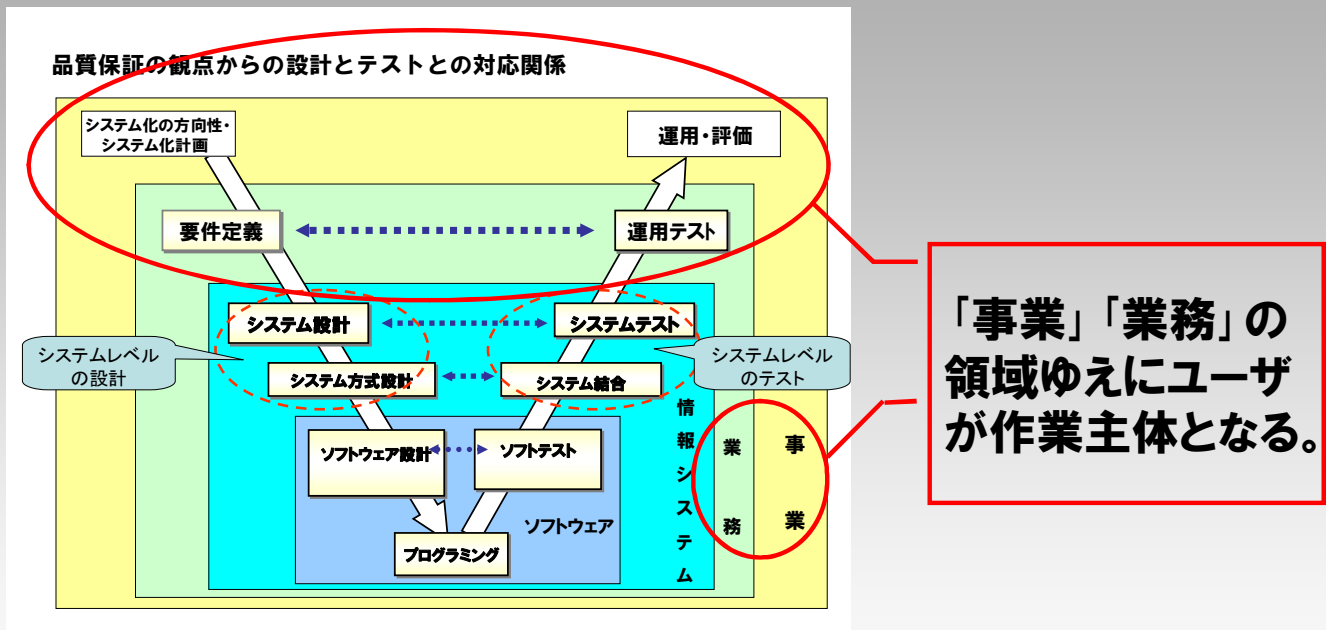
契約における役割分担の明示

外部設計書作成業務(準委任型)、ソフトウェア開発業務(請負型)、ソフトウェア運用準備・移行支援業務(準委任型)の場合のサンプル

項 番	工 程	作 業 内 容	作 業 分 担 (例)		
			ユーザ	ベンダ	備考
32	(前より続く)	共通機能設計		○	
33		部品プロセス設計		○	
34		移行プログラム設計		○	
35		中間ファイル・内部コード設計		○	
36		キャパシティプランニング		○	
37	システム設計 (内部設計)	開発環境の導入・設定(設備・ネットワーク等)	○(主体)	△(支援)	
38		スケジュール管理		○	
39		ユーザテスト体制・スケジュールの作成	○(主体)	△(支援)	
40		ベンダテスト体制・スケジュールの作成		○	
41		テスト計画レビュー	○(主体)	△(助言)	
42		移行レビュー	○(主体)	△(助言)	
43		開発協議会レビュー(内部設計関連)	○(主体)	○(主体)	
44			コンポーネントテスト仕様作成		○
45	ソフトウェア設計	プログラム構造設計		○	
46		プロセステスト仕様作成		○	
47		部品プログラム設計		○	
48		部品プロセステスト仕様作成		○	
49		移行プログラムテスト仕様作成		○	
50		部品プログラムコーディング		○	
51		部品プログラムテスト		○	
52		部品プログラムテスト仕様作成		○	
53		スケジュール管理		○	
54		プログラムテスト仕様作成		○	
55		部品プロセステスト		○	
56		移行プログラムテスト		○	
57		スケジュール管理		○	
61		移行アプリケーション機能テスト		○	

出典:2007年4月経済産業省報告書『情報システムの信頼性向上のための取引慣行・契約に関する研究会』
～情報システム・モデル取引・契約書～(受託開発(一部企画を含む)、保守運用)第一版』118～119頁

工程ごとの作業主体の傾向



出典：経済産業省報告書『「情報システムの信頼性向上のための取引慣行・契約に関する研究会」～情報システム・モデル取引・契約書～（受託開発（一部企画を含む）、保守運用）（第一版）』31頁、2007年4月

開発工程と作業主体

開発工程	ユーザ	ベンダ
システム化の方向性	○	
システム化計画	○	
要件定義	○	
システム設計 (システム外部設計)	○※	○※
システム方式設計 (システム内部設計)		○
ソフトウェア設計		○
プログラミング		○
ソフトウェアテスト		○
システム結合		○
システムテスト	○※	○※
運用テスト	○	

※ユーザまたはベンダのいずれかを示す。

ユーザ主体の場合のベンダの役割

ユーザが主体となって行う作業が技術的に適切に進み、当該作業のアウトプットが適切に纏まるよう必要な技術的なアドバイス、指導などの支援を行う。

このような支援を依頼する契約 ≡ **準委任契約**

(参考)

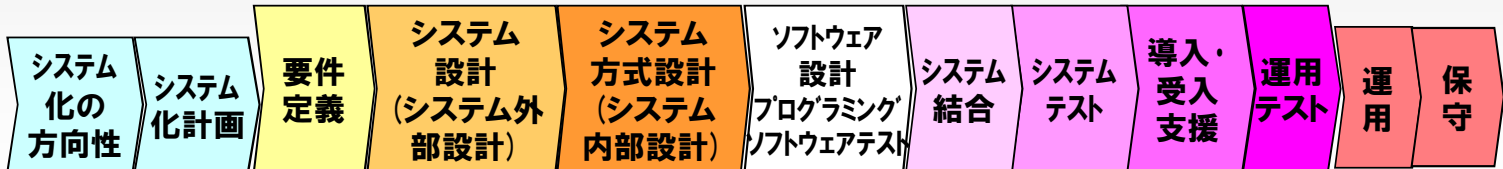
ベンダに仕事の完成を依頼し、
ベンダが主体作業を行う契約 ≡ **請負契約**

JEITAモデル契約について

	項目	JEITAモデル契約	経産省モデル契約
1	名称	ソフトウェア開発基本契約書	ソフトウェア開発委託基本モデル契約書
2	公表時期	08.10	07.4(第1版),08.4(追補版)
3	作成者	JEITAソリューションサービス事業委員会 ソフト開発モデル契約WGメンバー	経産省「情報システムの信頼性の向上のための取引慣行・契約に関する研究会」参加者 (ユーザ、ベンダ、弁護士他)
4	位置づけ	下記の点から 経産省モデル契約(07.4(第1版))を補整して作成 ・ソフトウェア開発経験を踏まえた検討の深化 ・ベンダの立場からの変更・選択	リファレンスモデルとしての活用を期待

JEITAモデル契約について

項目	JEITAモデル契約	経産省モデル契約
1 契約当事者	対等の交渉力のあるユーザとベンダ	
2 開発モデル	ウォーターフォールモデル	
3 対象システム	重要インフラ、企業基幹システムの受託開発(一部企画を含む)※第三者パッケージを購入、カスタマイズ開発する場合(追補版)、ハードウェア・プログラム製品の売買については対象外。	
4 開発プロセス	共通フレーム2007の要件定義段階、開発段階	同左 ※運用段階、保守段階について別モデル契約あり
5 その他	一括発注、工程により異なるベンダに発注する形態、マルチベンダ形態に対応	



← JEITA/経産省モデル契約の射程 →