

# ソフトウェア開発論

## 外部設計局面

第2.16版

木馬亭

□ 開発方針の立て方

□ DOA/その1

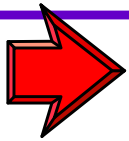
群馬県吾妻郡長野原町  
北軽井沢1990  
TEL 0279-84-5008  
2007-08-15 14:03  
000005

セットメニュー	¥3,600
セットメニュー	¥3,900
内税対象計	¥7,500
内税 5.0%	¥357
合計	¥7,500
お預り	¥10,000
お釣	¥2,500

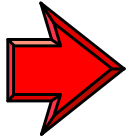
## ソフトウェア開発論・スケジュール

- はじめに
- 要求分析
- ➡ ■ 外部設計局面
- 内部設計
- プログラミング
- 単体試験
- 結合試験
- 総合試験
- 信頼性評価
- まとめ

## 外部設計局面／内部設計局面



- ステップ1: 要求分析の結果を基に、新ソフト開発方針を固める。



- ステップ2: 開発方針を基に、利用者、連携システムから見た形で、新ソフトの出来上がりを整理、設計し、キーマン、利用者に説明する  
(外部設計書の作成)。

- ステップ3: 外部設計を基に、実際の運用環境情報を付加し、内部構造を設計する。  
(内部設計書の作成)

## 外部設計の概要

方法論の底に流れている考えを理解しよう。

# 今から開発するソフトウェアを「擬人化」

- 人と、私(新たに開発するソフトウェア)間の対話に注目する。
- 人や、擬人化された人を「オブジェクト」と呼ぶ。
- 既存のソフトウェアや倉庫などのシステムも擬人化する。
- そうすると、世の中はオブジェクト間の**対話**として記述される。
- これを**オブジェクト指向設計**と言う。

注:ここでいう「私」「人」は、ある時、ある場所での、ある役割(role)を行っている人達、システムを総称している。  
たとえば山田さんは、とあるスーパーで平日9時から5時までは「従業員」として働いているが、帰宅時(5時半)にそのスーパーで買い物をする「客」として振る舞う。

## (お客さまとの)対話データは安定している。

- 木馬亭では対話データは
  - お客さま:ハンバーグ定食とステーキ定食
  - 接客係:¥7500
- 茨城大学学生食堂では対話データは
  - 学生:トンカツ定食
  - レジ係:¥600

### 木馬亭

群馬県吾妻郡長野原町  
北軽井沢1990  
TEL. 0279-84-5008  
2007-08-15 14:03  
000005

セットメニュー	¥3,600
セットメニュー	¥3,900
内税対象計	¥7,500
税	5.0% ¥357
合計	¥7,857
お預り	¥10,000
お釣	¥2,143

### ■ 縄文時代

- 山の民:桜貝2個ほしい。
- 海の民:ドングリ100個と交換しよう。

## (お客さまとの)対話プロセスは不定である。

- 木馬亭では対話プロセスは
  - お客さま:料理を注文する。
  - 接客係:料理を運ぶ。
  - 接客係:料金を受け取る。
- 茨城大学学生食堂では
  - 学生:料理を取る、注文する。
  - 学生:料理を運ぶ。
  - レジ係:料金を受け取る。
- 縄文時代は
  - 山の民が、海の民に希望する。
  - 物々交換のレートを協議する。
  - 物々交換する。
- 日立駅蕎麦では
  - お客さま:販売機で食券購入
  - そば屋のおばさん:蕎麦提供

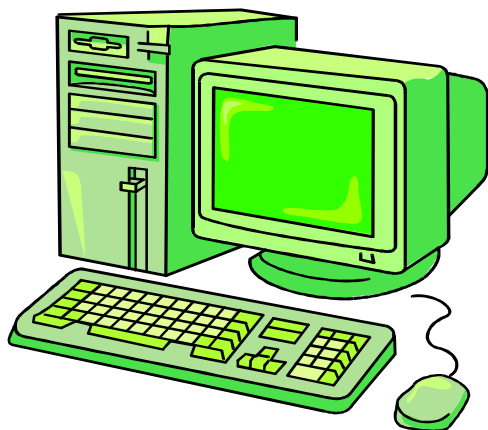
## 私(新たに開発するソフトウェア)は何者？

- 私は、特定機能を持った、対話データと対話プロセスの管理者
- 人間の代わりに、特定機能を実施する。
- **目的(何を管理するか)**は、ソフトウェア開発の最初に、開発方針で定義する。
- 木馬亭、私(注文管理システム)の例:
  - 料理の材料発注(毎日)、売り上げ(月次)管理、月別来客組数把握を**目的に**
  - (**管理に必要な**)対話データ:来客組ごとに、売り上げた料理名数と、それによる売り上げを記録、管理する。
  - 対話プロセス:
    - ▶ お客さま:料理を注文する。
    - ▶ 私:料理名数と売り上げを、来客組ごとに記録する。

# 演習：データ(DBサーバ)中心と独立型PC

## ■ DBサーバ中心

- 何が進歩したのか？
- 何がちがうのか？



## ■ 独立型PCとは

- お客様との対話によって得られたデータは他のPCと共有しない。
- したがって、他部門、他支店、隣の人と仕事を共有しない。

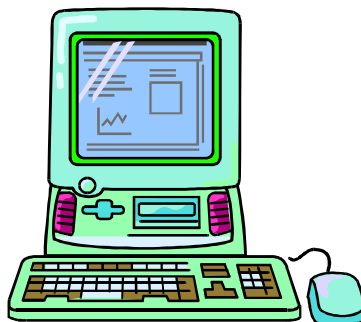
# データ(DBサーバ)中心と独立型PC

## ■ DBサーバ中心

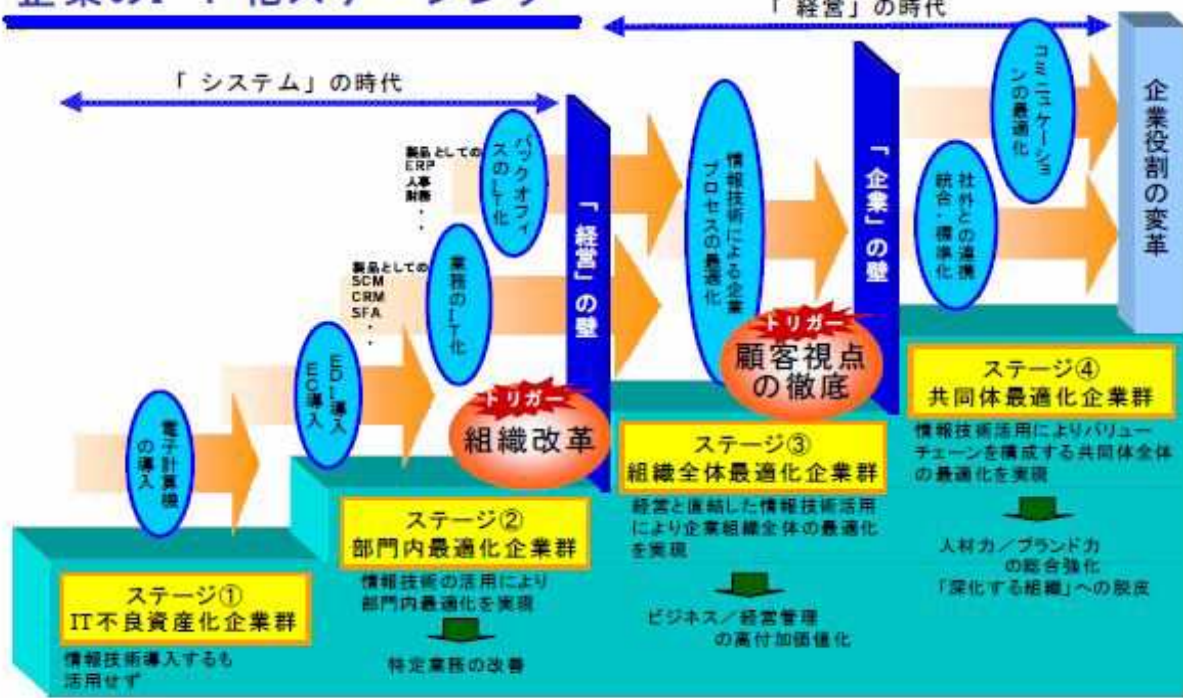
- データとプロセスの分離。データ共有
- それによる高度化(長所)
  - ▶ データの再利用が可能になる→
  - ▶ いろいろな部門による、多角的な分析が可能になる。
- 考慮点(課題)
  - ▶ DBサーバがこけると、全社業務停止になる。
  - ▶ データの整合性、機密保護など多方面のことを考慮しなければいけなくなるために、ソフト開発費が高価になる。

## ■ 独立型PC

- 隣のPCとさえ、データを共有できない! →
  - ▶ 仕事を共有できない!
  - ▶ したがって仕事が遅い!

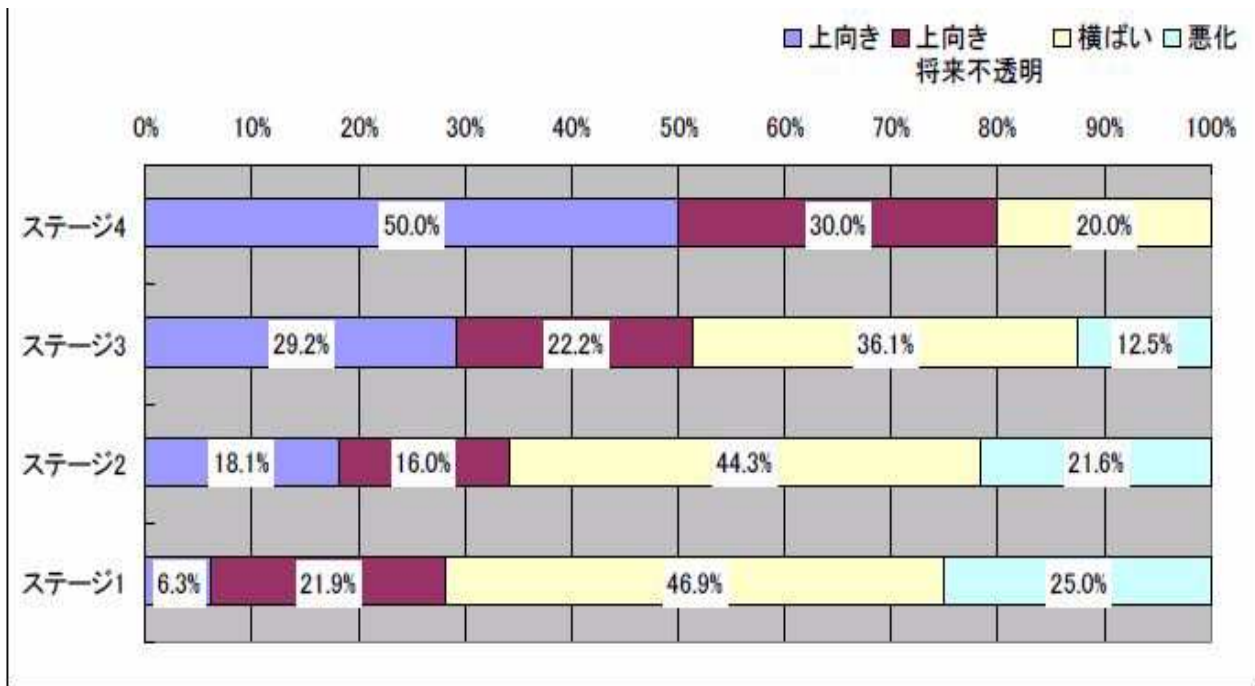


# 企業のIT化ステージング



(出所) 経済産業省「情報技術と経営戦略会議」

## ステージ別業況

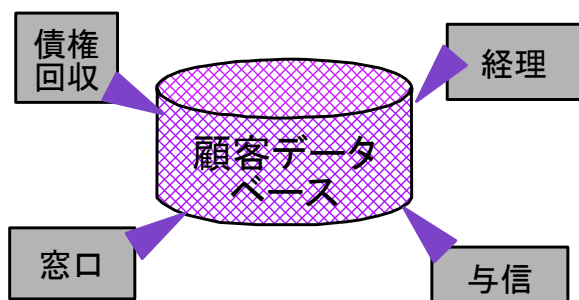


(ヒアリング及びアンケートによる経営者の業況認識による。)

# A社におけるデータ(DBサーバ)中心の採用

## ■ 今後(データ中心採用)

- 全社共通の顧客データベースを業務活動の中心にすえる。
- すべての部門が部門固有の業務活動を、データベースと対話する形で記述する。



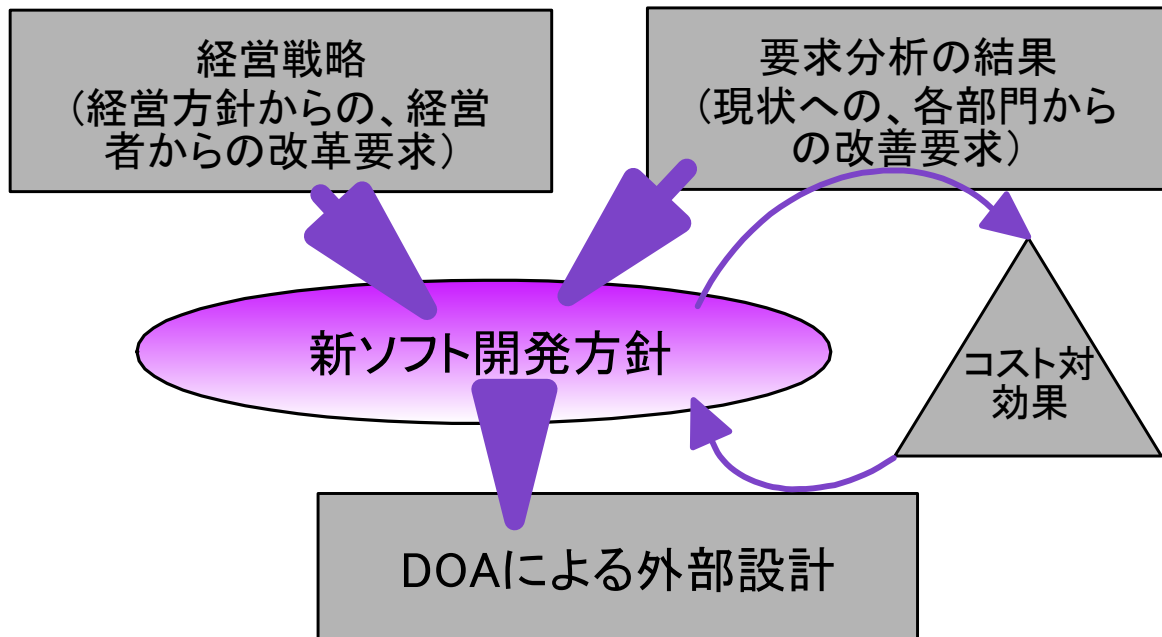
## ■ 従来(独立型PCベース)

- 部門(窓口、与信、債権回収、経理...)別に、別々のプログラムが格納されたPCを、個人別に配給
- その結果、お客様との対話データを引き継ぐには、PCからPCへの転送が必要
- ましてや、再訪問されたときの、前回データの再利用などは...
- 他支店へ訪問されでもしたら、もうお手上げ...

## 外部設計

いよいよ正式な外部設計を開始する。

# ステップ1：要求分析の結果を基に、 新ソフト開発方針を固める。

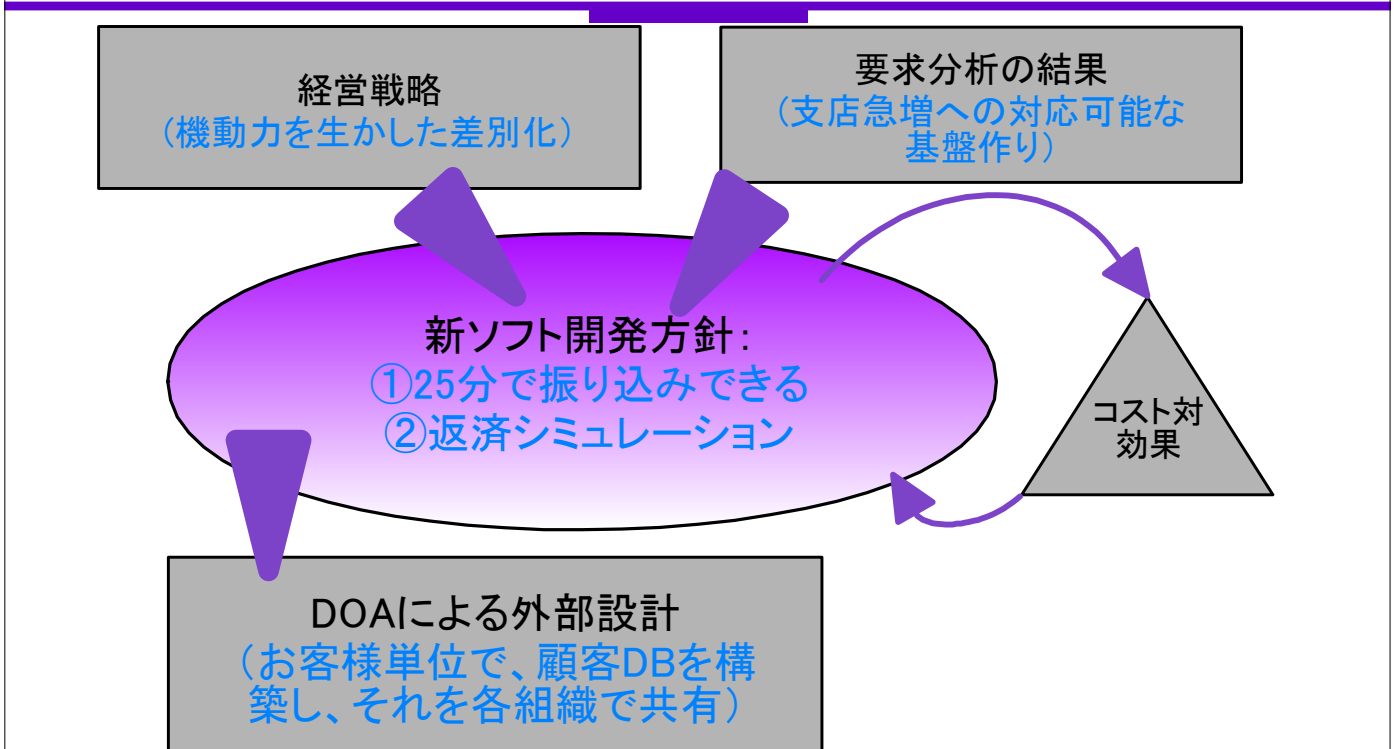


## 各社経営戦略

- 業績好調A社の場合：支店を、毎年20%の割合で増やす。それに対して人員増でなく、設備投資で対処したい。
- 業績不調B社の場合：支店は増やさず、人員を3割カットすることで、赤字解消を図りたい。設備投資も前年比7割カットで行きたい。
- 堅実路線C社の場合：支店数は業界伸び率平均にとどめ、利益率を現状の2%から、3年間かけて4%まで引き上げたい。そのための施策は積極的におこなう。



# ステップ1の例：A社の新ソフト開発方針決定



## 要求分析－

## 返済シミュレーションの場合

利用者(WHO)	業務範囲(WHAT)	業務実施方法(HOW)	楽にしたいもの＝新ソフト開発の動機(WHY)
経理部門麻生さん	顧客DBより、毎月の利子をコンピュータが計算したものを基に、顧客への請求業務	1. 年利を月歩になおして、複利計算。1円未満は切り上げ 2. 延滞者への督促	1. 多種の商品対応が可能になるようにしてほしい。 2. 日歩でも計算出来るようにしてほしい。
計画部門佐久間さん	短期資金繰り計画および長期投資計画	1. 請求書総額、延滞金総額、貸出金総額を基に、来月の必要資金を手計算で予測 2. 昨年の決算を基に、手計算で5年後までの投資計画を策定	1. 予測作業をコンピュータ化し、毎日修正出来ないか。 2. 投資計画を、現実の顧客DBに基づいて作成できないか。
窓口(第一線)林さん	顧客借入相談、与信	1. 顧客からの要望した貸し出しに対する利子を含めた返済計画表を、電卓で作成する。 2. 問題なければ与信可能かどうか、審査部門へ申請をFAXで行う。	1. 相談が多く、負荷が高い。計画表とFAXについて楽に出来る方法は無いか。 2. 多忙で計算間違いが増えている。防ぐ方法は無いか。
借金をしたい人・人僱さん?	借入、返済	1. 窓口へ行くか無限さんを利用して、借入計画を立てることもある。 2. 与信が得られれば融資を受ける。 3. 毎月、返済をATMか銀行から行う。	1. A社は、返済計画表をお願いすると、時間がかかりすぎてたまらない。

## 折り合いの付け方

- 方法論の決定打は、現在のところない。
- 方法論とまでは行かないが、多くの人が言っていること
  - キーマン(この人をウンと言わせれば、開発方針が決まる)を探せ！
    - ▶ キーマンは、中小企業では社長のことが多い。いわゆるワンマン・タイプの企業
    - ▶ 大企業では、ワンマン・タイプ企業がある一方で、各部門に隠れたキーマンがいるタイプの企業が多い。
  - キーマンがウンという案を出すように努力する。
  - キーマンが、部門代表の場合は、全体(消費者)を忘れた議論に走り勝ちなので、注意しながら議論をリードする。

## A社における折り合いの付け方

- A社は、ワンマン社長だが、現場の声を重視している。
  - 現場の意見を十分に聞く。
  - その中で、戦略に沿ったものを、開発方針(案)として社長に相談する。
  - 承認を得たものを、開発方針として、設計を開始する。

### ■ 開発方針(案)

変更点	新業務実施方法	従来の業務実施方法	新業務の長所	新業務の課題
25分以内の振り込み可能とする、業務プロセスの実現	・顧客DB中心への全社業務プロセス変更	各部門連絡は、FAXもしくは電話。部門内は口頭もしくは紙ベース連絡	最短の振り込み時間実現による競争力強化、他社は翌日	・業務プロセス改革のため、要全社的調整 ・要業務新プロセス徹底
多部門からの利用可能な、顧客DB中心返済シミュレーション・ソフトの提供	・窓口、経理、与信、債権回収、長期計画部門共通機能提供 ・既存借入との合計回答サービス開始	・窓口、与信は電卓、経理はコンピュータ、その他は山勘 ・窓口は、新規借り入れのみ電卓で計算	・与信精度の向上 ・借りすぎ防止 ・無限さん導入、顧客自身の操作による利便性向上	利子変動への対応のために、顧客DBの再編成要。 今回は見送り

## 演習（外部設計・開発方針）

- A社の例を参考に、自分が今から開発するソフトウェアについて開発方針を立ててみましょう。
- たとえば、A社開発方針を少し変えてみます。
- 全く違う、独自の開発方針でも構いません。



## ステップ2：外部設計書作成

開発方針

外部設計書：  
開発方針を基に、利用者、連携システムから見た形で、新ソフトの出来上がりを整理、設計し、キーマン・利用者に説明する。

内部設計書

# 外部設計書作成の流れ

木馬亭

群馬県吾妻郡長野原町  
北軽井沢1990  
TEL. 0279-84-5008

2007-08-15 14:03  
000005

セットメニュー	¥3,600
セットメニュー	¥3,900
内税対象計	¥7,500
内税 5.0%	¥357
合計	¥7,857
お預り	¥10,000
お釣	¥2,143

①データ保管形式の設計  
(対話データ管理の設計)

②データを呼び出し操作する  
部分の機能設計  
(対話データを使った、  
対話プロセスの設計)

外部設計書

①来客組ごとに、注文  
を受ける。  
②食事後、対価を請求  
する。

木馬亭の接客業務

## 外部設計書（仕様書）とは？

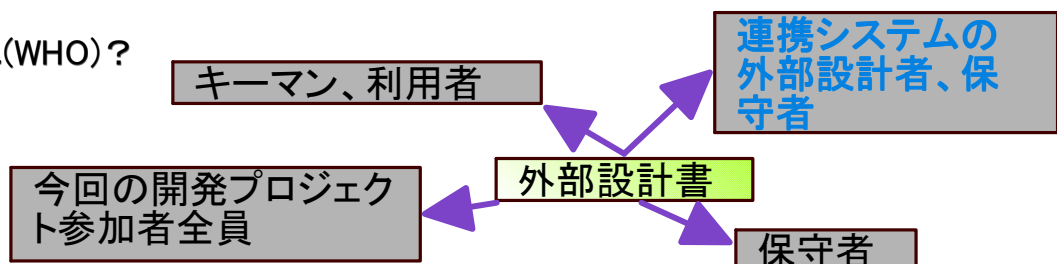
### ■ 何を作るの(WHAT)？

- これから開発するソフトは何なのか、利用者、**連携システム**が機能进行操作するのに必要な説明(外部設計)を記述する。HOWは内部設計、それ以外は外部設計と考えても良い。

### ■ 何のために(WHY)？

- キーマン、利用者に対して、出来あがる外部設計を説明し、
  - キーマン、利用者イメージとすり合わせるにより、納品後のトラブルを避ける。
  - この後の変更要求は受け付けない口実にする。
- 内部設計書、結合試験テストケース、プロジェクト・スケジュール、開発体制は、外部設計書に基づき作成する。

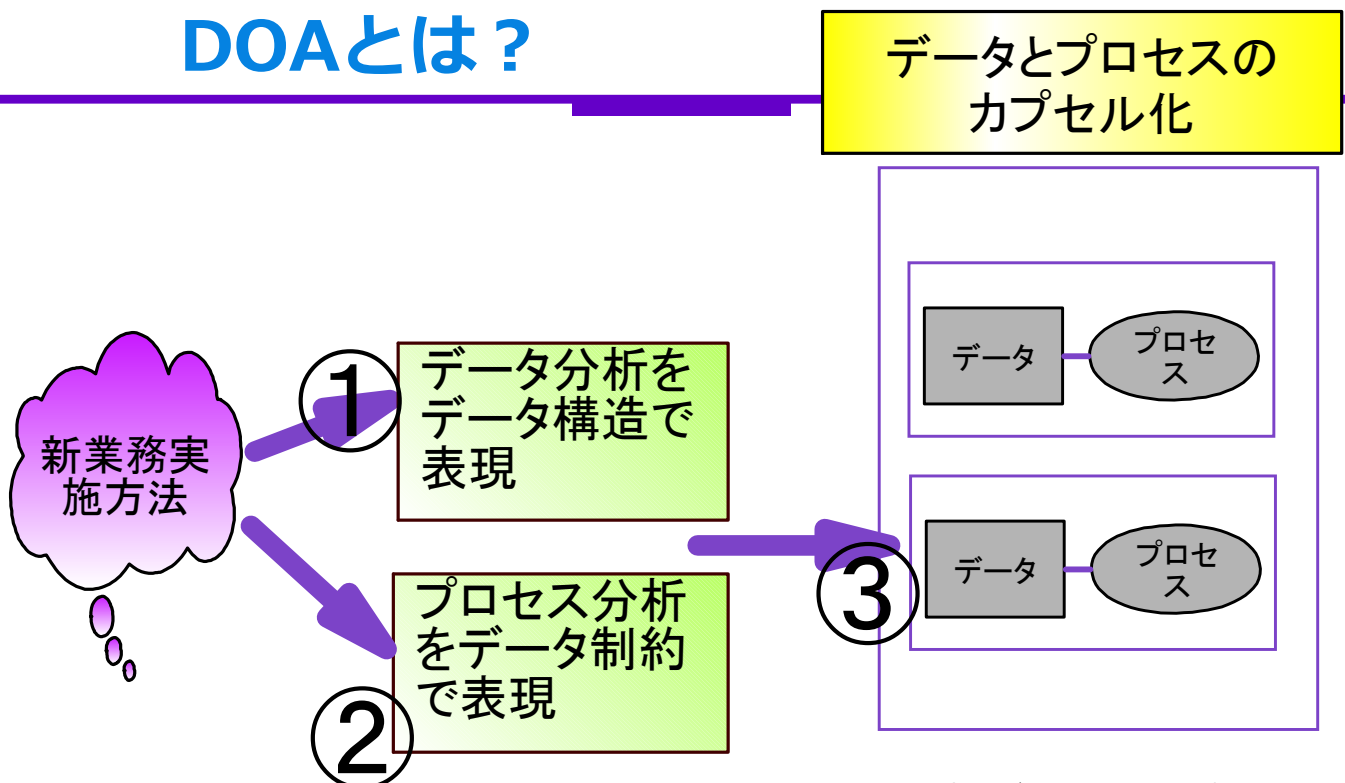
### ■ 誰のために(WHO)？



# 外部設計への取り組み

- ここでは、DOA(データ中心アプローチ)に基づき設計してみる。
  - まず、外部設計をおこなう。
    - ▶ 最初に人と私の間の対話データから、管理の必要なデータを抜き出し、それを中心にデータ構造を分析する。
    - ▶ つぎに、そのデータを使用した処理=プロセスを抜き出し、それをデータ制約という見方でまとめる。
  - つぎに、内部設計をおこなう。
- DOAの特徴
  - まず対話データから先に分析する。
  - プロセスを単に、データの変換機能=「データ制約」と位置付ける。
- なぜDOAか？
  - 対話データのほうが、多くの場合安定している(変化が少ない)ためである。

## DOAとは？



出典:IRA研究会、データ中心システム分析と設計

# DOAと局面の関係

## ③は内部設計

データとプロセスの  
カプセル化

## ①②は外部設計



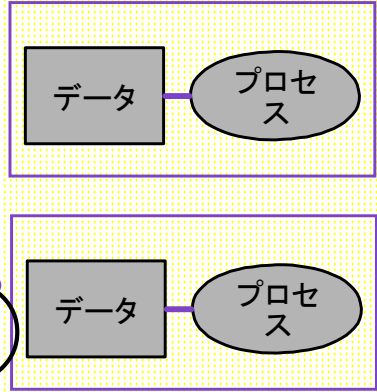
①

データ分析を「データ構造」(表形式)で表現。→DFD表現、ER図などで業務分析

②

プロセス分析を「データ制約」(表形式)で表現

③



出典:IRA研究会、データ中心システム分析と設計

# 開発方針・木馬亭注文管理システムの例

変更点	新業務実施方法	従来の業務実施方法	新業務の長所	新業務の課題
料理の材料発注(毎日)、売り上げ(月次)管理、月別来客組数把握	<ul style="list-style-type: none"> <li>来客組ごとに、注文料理名数と、それによる売り上げを記録、管理</li> <li>それを基にきめこまかい管理</li> </ul>	<ul style="list-style-type: none"> <li>材料置き場の目検による材料発注(ランダム)</li> <li>レジスターによる売り上げ管理(月次)</li> </ul>	<ul style="list-style-type: none"> <li>きめ細かい材料発注→新鮮な材料使用→評価上昇</li> <li>経理システムとの売り上げ、原価計算連動→より速い経営判断</li> </ul>	<ul style="list-style-type: none"> <li>支店が少ないため、現時点では投資コスト負担大</li> <li>投資回収には4年が必要</li> </ul>

2007-08- 4:03  
000005

セットメニュー	¥3,600
セットメニュー	¥3,900
内税対象計	¥7,500
内税 5.0%	¥357
合計	¥7,500
お預り	¥10,000
お釣	¥2,500

これが重要

# 注文管理システム - データ分析

私(注文管理システム)からみて、

- 私と対話する(よってくる)人、システムは誰?
- 人、システムごとに、どんなときに、どんな頻度でよってくる?
- 対話はどんなデータを使って、どんなプロトコル(操作手順)でやられるか?
- 対話に使われるデータの詳細は何?

ソフト開発に必要な、外との 約束データ

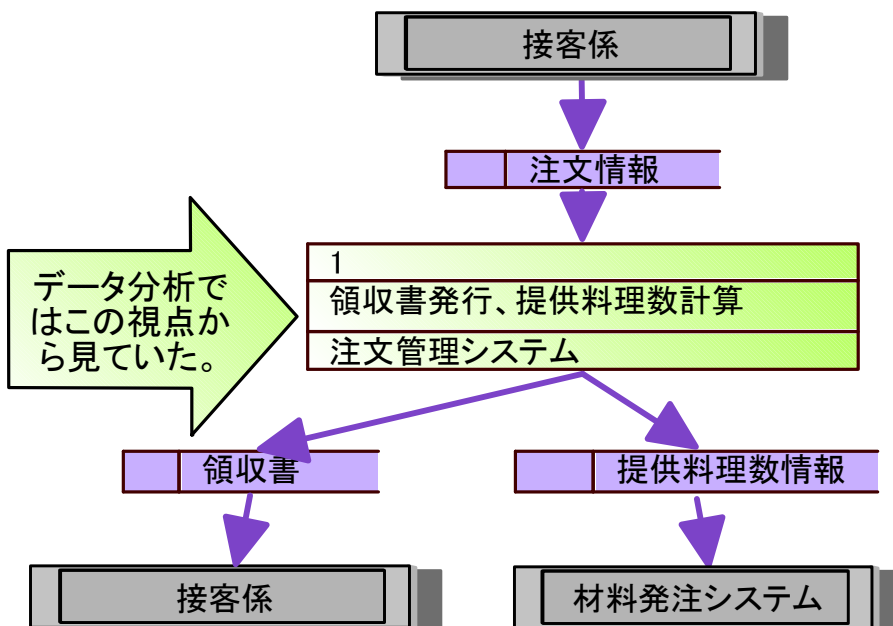
操作は上から下へ

WHO	WHEN(CONDITION)	WHAT(ACTION)	WHAT(DATA)
接客係	一日40組 (最大同時4組) /二店舗	<ul style="list-style-type: none"> <li>・お客さま: 料理を注文する。</li> <li>・接客係: 料理名数と売り上げを来客組ごとに記録する。</li> <li>・私: 領収書を発行する。</li> </ul>	注文情報 { 来客組ごとに、提供した料理名、数と、それによる売り上げ }
材料発注システム	一日一回	<ul style="list-style-type: none"> <li>・材料発注システム: 問い合わせをかける。</li> <li>・私: 当日、それまでの、料理名ごとの提供数を答える。</li> </ul>	提供料理数情報 { 当日、料理名ごとに、提供数 }

## 注文管理システム・業務分析

### - DFD表現

DFDでは、データの流れ、利用の仕方を把握する。ER図ではデータ間の関係を理解する。



データ分析ではこの視点から見ていた。

注:

- 外部エンティティ
- データストア
- 番号
- 処理機能
- 人、部門名
- データフロー

# 注文管理システム設計－データ制約

表	属性	ドメイン制約	主キー	存在制約	参照制約	多重度制約	導出制約	関連制約	更新制約	処理順序制約
A=	日付	{yyyy/mm/dd}, 年月日	YES	必須	なし	1	なし	なし	なし	なし
注文情報	来客組ID	{hh:mm}, 時刻+ { 001..999 }, 整数	YES	必須	なし	1	なし	なし	なし	来客順に振る。
	C=料理名	{ 2..20 }, 文字	YES	必須	F	1	なし	なし	なし	なし
	D=注文数	{ 1..99 }, 整数	NO	必須	非該当	1	なし	Cによる制限	なし	なし
	E=売り上げ	{0..999999}, 整数	NO	必須	非該当	1	なし	なし	D<=0 E=0	支払い時確定
B=提供料理数情報	F=料理名	{ 2..20 }, 文字	YES	必須	なし	1	なし	なし	なし	名前順にソート
	提供数	{ 1..999 }, 整数	NO	有効な主キーC,D存在	非該当	1	当日C別D合計	なし	なし	A更新時以外に計算

## 参考：データ分析の結果

WHO	WHEN(CONDITION)	WHAT(ACTION)	WHAT(DATA)
接客係	一日40組 (最大同時4組)/二店舗	<ul style="list-style-type: none"> <li>お客さま：料理を注文する。</li> <li>接客係：料理名数と売り上げを来客組ごとに記録する。</li> <li>私：領収書を発行する。</li> </ul>	注文情報 [ 来客組ごとに、提供した料理名、数と、それによる売り上げ ]
材料発注システム	一日一回	<ul style="list-style-type: none"> <li>材料発注システム：問い合わせをかける。</li> <li>私：当日、それまでの、料理名ごとの提供数を答える。</li> </ul>	提供料理数情報 [ 当日、料理名ごとに、提供数 ]

## 配布資料

- データ制約、各制約の意味説明
  - 木馬亭、注文管理システムを例にして
- 外部設計付録



## 演習（外部設計）

- 返済シミュレーション・ソフトの場合について、外部設計してみましょう。



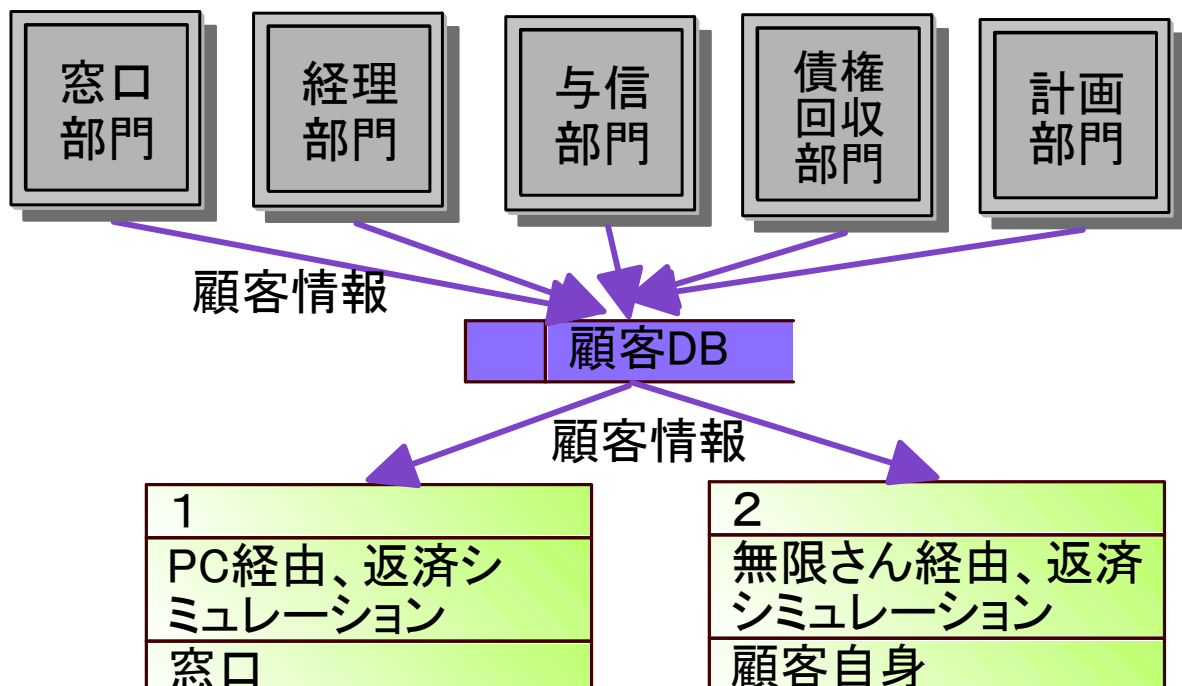
## A社開発方針例

変更点	新業務実施方法	従来の業務実施方法	新業務の長所	新業務の課題
25分以内の振り込み可能とする、業務プロセスの実現	・顧客DB中心への全社業務プロセス変更	各部門連絡は、FAXもしくは電話。部門内は口頭もしくは紙ベース連絡	最短の振り込み時間実現による競争力強化、他社は翌日	・業務プロセス改革のため、要全社的調整 ・要業務新プロセス徹底
多部門からの利用可能な、顧客DB中心返済シミュレーション・ソフトの提供	・窓口、経理、与信、債権回収、長期計画部門共通機能提供 ・既存借入との合計回答サービス開始	・窓口、与信は電卓、経理はコンピュータ、その他は山勘 ・窓口は、新規借り入れのみ電卓で計算	・与信精度の向上 ・借りすぎ防止 ・無限さん導入、顧客自身の操作による利便性向上	利子変動への対応のために、顧客DBの再編成要。今回は見送り

## A社データ分析例

WHO	WHEN(CONDITION)	WHAT(ACTION)	WHAT(DATA)
窓口	ランダムに顧客が100人/台/日	窓口PCに入力し、印刷物を手渡し	顧客DB { 顧客ID、 本日の金利、 返済予定日数、 元金、 返済額 }
顧客自身	ランダムに顧客が50人/台/日	「無限さん」に顧客自身で入力し、画面に表示	上に同じ

## A社業務分析 – DFD表現



# A社データ制約

データ制約の種類	ドメイン制約	識別子(主キー)	存在制約	参照制約	多重度制約	導出制約	関連制約	更新制約	処理順序制約
A=顧客ID	{ 1000000 .. 9100000 }, 整数	YES。顧客DBを通してユニークであること	必須	参照不可	1	なし	なし	なし	見積もり後、顧客が望む場合は与信システムに処理を引き継ぐ
B=本日の金利	{ <0%..23% }, 整数もしくは実数	NO	必須	非該当	1	なし	なし	なし	なし
C=返済予定日数	{ 0..3650 }, 整数	NO	必須	非該当	1	なし	なし	なし	なし
E=返済額	{ 0.8桁 }, 正の整数	NO	有効なA,B,C存在	非該当	1	$E = (1 + B/36500) ** C$ 1円未満は切り上げ	なし	なし	なし

返済シミュレーション・ソフトで導出(計算)

## おわり