

# 事例3：要件定義ドキュメントの漏れや 曖昧性の低減

IPAセミナー@東京  
2019年2月26日

独立行政法人情報処理推進機構（IPA）  
社会基盤センター 産業プラットフォーム部

システム化要求WG委員  
株式会社NTTデータ  
崎山 直洋

## ■第4章

# 要件定義成果物の品質向上

● 4.1 成果物の作成計画時の留意点

※解説する部分

● 4.2 主要な成果物と作成上の留意点

● 4.3 成果物に共通の留意点

※解説する部分

## 4.1 成果物の作成計画時の留意点

- 4.1.1 成果物を決めるための勘どころ
- 4.1.2 成果物作成上の役割分担を決めるための勘どころ

※解説する部分

## 4.2 主要な成果物と作成上の留意点

- 4.2.1 ビジネスプロセス関連図
- 4.2.2 業務機能構成表
- 4.2.3 ビジネスプロセスフロー（業務フロー／システム化業務フロー）
- 4.2.4 画面／帳票レイアウト
- 4.2.5 業務処理定義書
- 4.2.6 概念データモデル（ER図）
- 4.2.7 エンティティ定義書／データ項目定義書
- 4.2.8 CRUD図
- 4.2.9 総合テスト計画書
- 4.2.10 システム移行計画書
- 4.2.11 運用・操作要件書
- 4.2.12 非機能要件書

## 4.3 成果物に共通の留意点

- 4.3.1 要件定義成果物のレビュー
- 4.3.2 要件定義における未決定事項の取扱い
- 4.3.3 要件定義文章の品質向上

※解説する部分

## ■ 第4章で説明している事項

### ● 個々のドキュメントの視点で要件定義の勘どころを紹介する

#### □ 4.1 成果物の作成計画時の留意点

**要件定義成果物を決める**際の注意点を紹介する

※解説する部分

#### □ 4.2 主要な成果物と作成上の留意点

各ドキュメントの作成上で想定される課題の解決や品質向上のための勘どころについて、今回は設計工程に直接つながる**代表的な12のドキュメント**を選んで紹介する

#### □ 4.3 成果物に共通の留意点

ドキュメントに**共通**の勘どころを説明する

※解説する部分

## ■ 4.1 成果物の作成計画時の留意点

- 4.1.1 成果物を決めるための勘どころ
  - コラム 成果物に記載する範囲を決める
  - コラム フロー系成果物のプロセスの粒度を決める
- 4.1.2 成果物作成上の役割分担を決めるための勘どころ

# 4.1.1 成果物を決めるための勘どころ (1)

企画プロセス		要件定義プロセス	システム要件定義プロセス (外部設計：※合意形成ガイド)
構想立案	システム化計画立案		
問題点、課題、ニーズなど一覧	問題点、課題、ニーズなど一覧	問題点、課題、ニーズなど一覧	
要求一覧	要求一覧	要求一覧	
業務機能構成表	業務機能構成表	業務機能構成表	システム化業務一覧
ビジネスプロセス関連図	ビジネスプロセス関連図	ビジネスプロセス関連図	
業務フロー	業務フロー	業務フロー	
システム化業務フロー	システム化業務フロー	システム化業務フロー	システム化業務フロー
	業務処理定義書	業務処理定義書	
		システム化業務説明	システム化業務説明
		画面一覧	画面一覧
		画面遷移	画面遷移
		画面レイアウト	画面レイアウト
		画面入出力項目一覧	画面入出力項目一覧
概念データモデル(ER図)	概念データモデル (ER図)	概念データモデル (ER図)	
		エンティティ一覧	エンティティ一覧
		エンティティ定義書	エンティティ定義書
		CRUD図	CRUD図
外部システム関連図	外部システム関連図	外部システム関連図	外部システム関連図
		トメニュー一覧/定義	トメニュー一覧/定義
		コード一覧/定義	コード一覧/定義



前後のプロセスの作業・成果物との関係を明確にする  
 ↓  
 要件定義における各成果物の目的がより具体化される  
 ↓  
 記載範囲・深さが定まる

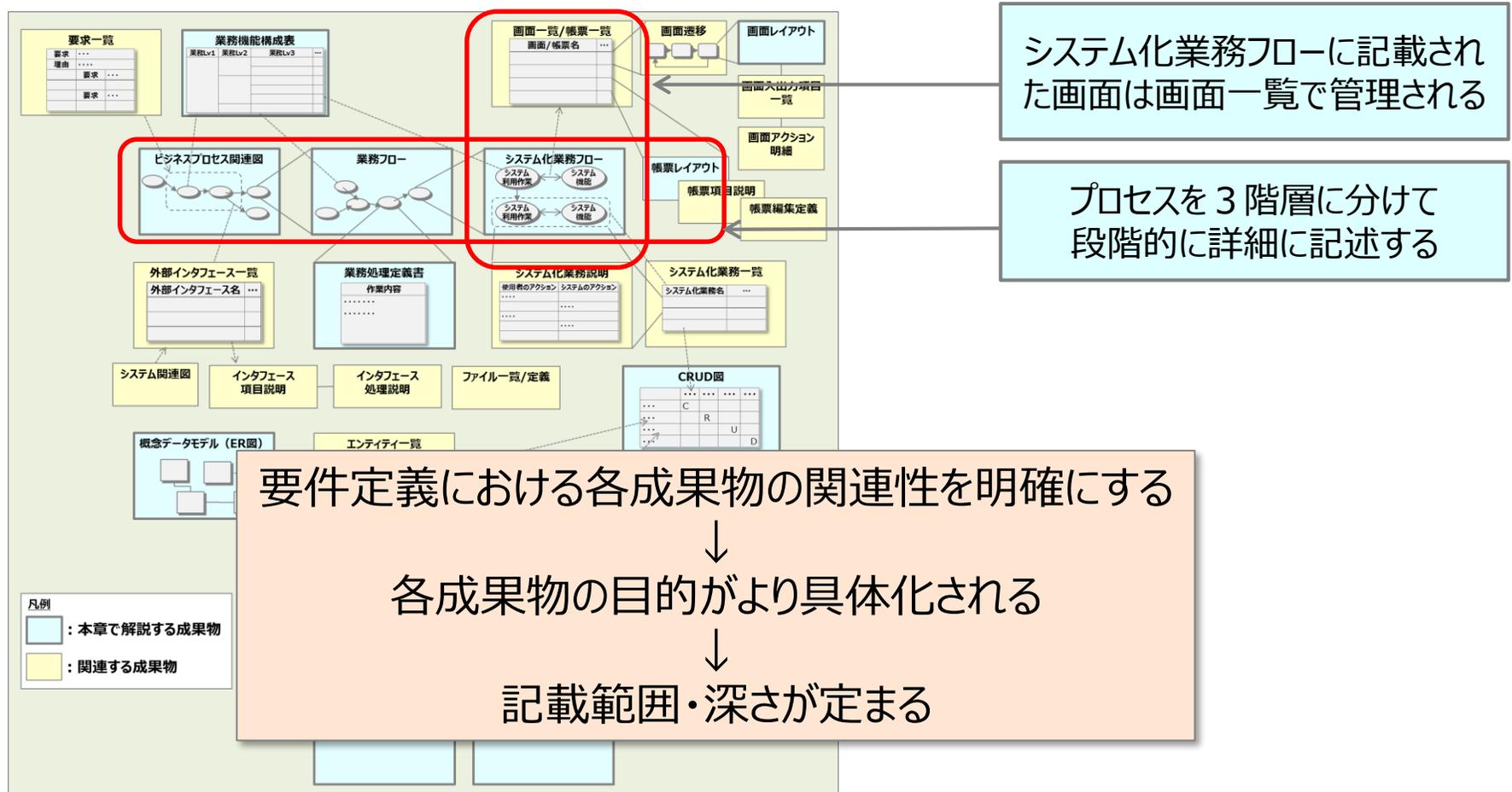
※左記は要件定義において作成される主要なドキュメントであるが、これがすべてではない

※どのような条件のシステム開発の要件定義を行うかによって、要件定義のスタイルもドキュメントも変化する

前後のプロセスの目的を踏まえて、プロセスに成果物をマッピングすべし

# 4.1.1 成果物を決めるための勘どころ (2)

## ■ ドキュメント関連図のイメージ



**成果物全体の関連を明確にすべし**

## 4.1.1 成果物を決めるための勘どころ (3)

### ■ 要件定義の成果物を決める際の注意点

- 要件定義の各成果物は**そのプロジェクト**の置かれている状況や目的を考慮して決める
- 要件定義は、システム開発に**仕様を伝えるだけでない**ことを踏まえて成果物を選択する (次頁)
- 選ばれた**成果物の目的**を明確にする

プロジェクトの状況や利用目的を意識して成果物を決定すべし

# 補足 1 2 以外のドキュメントの考慮

## ■ 要求仕様を設計開発につなげる

1 2 ドキュメント

要求仕様を表現する図表（状態遷移図など5.8参照）

## ■ 要求を分析する

問題点原因分析図（3.1参照）

要求構造図（3.1参照）

ステークホルダ関連図（3.5参照）

業務シナリオマトリクス（5.3参照）

## ■ 理解を促進する

業務用語定義

バリエーション定義（3.3参照）

## ■ 合意形成を行う

Before-After図（上層部に1枚で変化や効果を説明するための資料）

.....

課題：「要件定義の抜け、漏れが問題だ！」

→全ての成果物を抜け漏れ無く書かなければならないのか

対策：成果物が「**主要**」で良いのか「**すべて**」でなければならぬのかを記載項目ごとに明確にしておく

- ① 一覧系成果物の扱いを明示すべし
- ② どのプロセスで作成するかによって記載範囲が変わる成果物を明示すべし
- ③ 後続の成果物で記載を補完する場合のドキュメント関係を明示すべし

# 【コラム】～成果物に記載する範囲を決める～ (2)

## ■ 一覧系成果物は対象物を網羅するために作成される

- 漏れのチェックや変更管理に適する ⇒ 「**完全性**」が求められる
  - 何を管理するのかを明確にし、他の成果物との関係を明示する
    - 「問題点一覧（すべてを記載）」←→「問題点原因分析図（図で分析が必要な複雑な問題）」
    - 「業務機能構成表のプロセス（すべてを記載）」←→「業務フロー上のプロセス（主要なもの）」
    - 「画面一覧/帳票一覧（処理単位に分割された画面名を記載）」
- ←→「システム化業務フローの画面（機能単位で集約された画面名）」

企画プロセス		要件定義プロセス	システム要件定義プロセス
構想立案	システム化計画立案		(外部設計：※合意形成ガイド)
問題点、課題、ニーズなど一覧	問題点、課題、ニーズなど一覧	問題点、課題、ニーズなど一覧	
要求一覧	要求一覧	要求一覧	
業務機能構成表	業務機能構成表	業務機能構成表	システム化業務一覧
		画面一覧	画面一覧
		エンティティ一覧	エンティティ一覧
	外部インターフェース一覧	外部インターフェース一覧	外部インターフェース一覧
		バッチ処理一覧	バッチ処理一覧
	帳票一覧	帳票一覧	帳票一覧
		ドメイン一覧/定義	ドメイン一覧/定義
		コード一覧/定義	コード一覧/定義

一覧系成果物の扱いを明示すべし

# 【コラム】～成果物に記載する範囲を決める～ (3)

- 概念データモデルなど、どのプロセスでも作成される可能性があるものは、「どのプロセスでは、どの範囲で記載するのか」を明示する

企画プロセス		要件定義プロセス	システム要件定義プロセス
構想立案	システム化計画立案		(外部設計：※合意形成ガイド)
概念データモデル(ER図)	概念データモデル (ER図)	概念データモデル (ER図)	E R 図
		エンティティ一覧	エンティティ一覧
		エンティティ定義書	エンティティ定義書
		C R U D 図	C R U D 図
	企画プロセス	要件定義プロセス	システム要件定義プロセス
エンティティ	主要エンティティ	全エンティティ	全エンティティ
データ項目	主要データ項目	主要データ項目	全データ項目

**どのプロセスで作成するかによって記載範囲が変わる成果物を明示すべし**

## ■ ドキュメントの完全性を分ける例

### ● 「業務フロー」

- 業務改革レベルの変更を行う業務だけを対象として作成する

### ● 「システム化業務フロー」

- システム化するすべての業務について記載する

## ■ 網羅性・完全性の追求は後続のドキュメントで実施する、 という判断もある

**後続の成果物で記載を補完する場合のドキュメント関係を  
明示すべし**

## 【コラム】～フロー系成果物のプロセスの粒度を決める～

課題：人によって、大まかになったり、やたら細かくなったり、業務フローのプロセスの粒度を揃えることに苦勞する

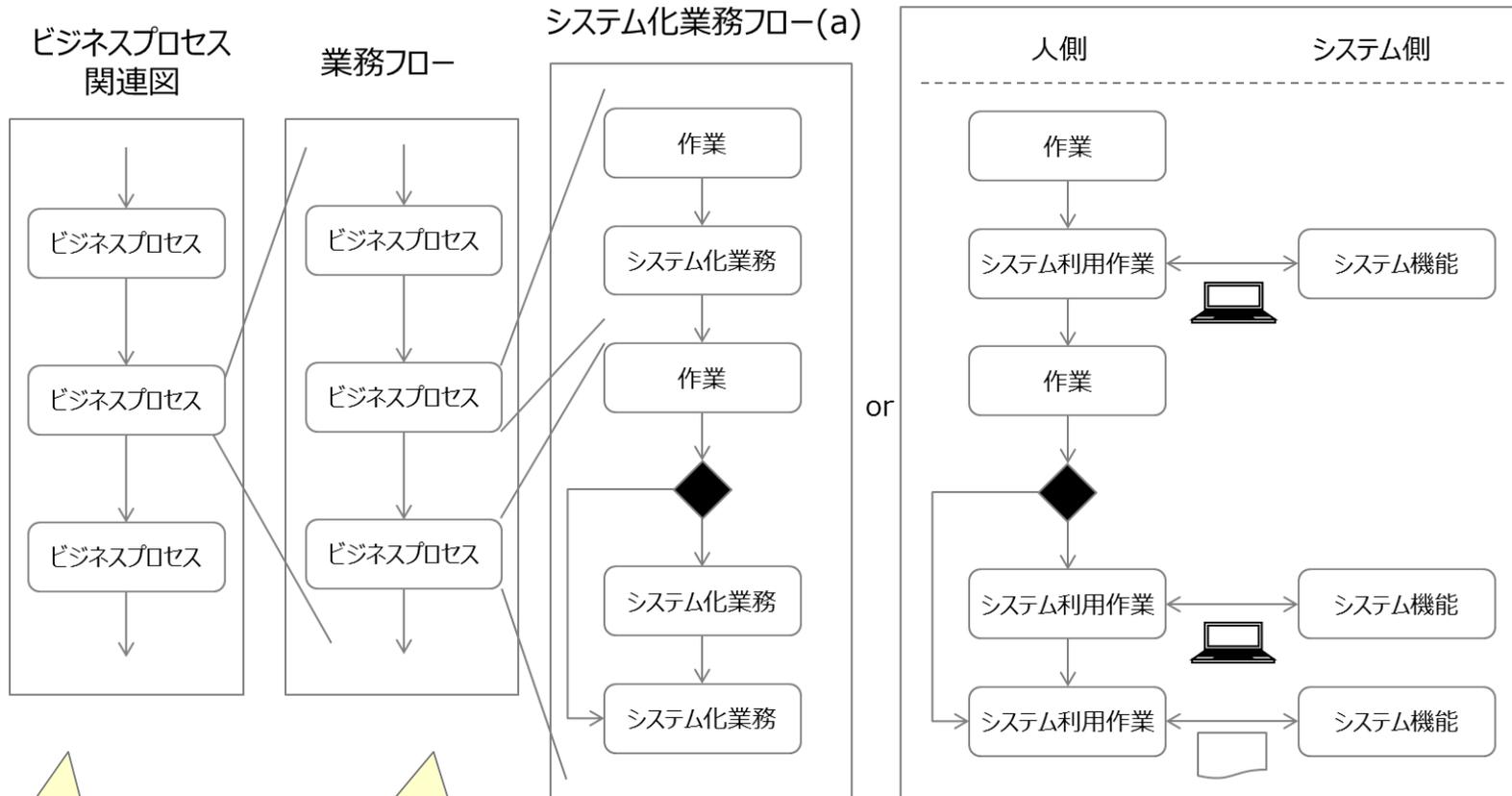
対策：**記述の粒度を定義**し、成果物と対応づけて書き方として提示することが重要

- 業務プロセス階層化のレベルを業務間で合わせる
- どの階層のプロセスをどのドキュメントに記載するかを規定する
- 作成上のルールを提示する
- ドキュメント記載の粒度の考え方を提示する

# 【コラム】～フロー系成果物のプロセスの粒度を決める～例

3階層で書く

名前を付けよ



一枚で書く

Whatで書く

人とシステムを区別して書く

- 成果物作成にあたっては、どの部門が**主体となって作業**するかを明確にする
  - 主体とは、あくまでドキュメント作成の主体であり、責任者ではない
- **要件定義は利用部門（オーナー部門）の責任**
  - SEC BOOKSにも記載の大原則  
「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～」
  - しかし、利用部門だけに要件定義を押しつけるのは現実的でない
  - 双方協力して実施するのが望ましい

### ■ 役割分担の例（作業主体）

項番	成果物	役割分担			
		ユーザ			ベンダ
		業務部門	システム部門	運用部門	
4.2.1	ビジネスプロセス関連図	○	△	△	△
4.2.2	業務機能構成表	○	△	△	△
4.2.3	ビジネスプロセスフロー (業務フロー／システム化業務フロー)	○	△	△	△
4.2.4	画面／帳票レイアウト	△	○	△	△
4.2.5	業務処理定義書	○	△	△	△
4.2.6	概念データモデル（ER図）	△	○	△	△
4.2.7	エンティティ定義書／データ項目定義書	△	○	△	△
4.2.8	CRUD図	△	○	△	△
4.2.9	総合テスト計画書	○	○	○	△
4.2.10	システム移行計画書	○	○	○	△
4.2.11	運用・操作要件書	○	○	○	△
4.2.12	非機能要件書	△	○	△	△

凡例) ○：作成主体、△：作成支援（情報提供、確認（チェック）、アドバイス）

## ■ 4.3 成果物に共通の留意点

- 4.3.1 要件定義成果物のレビュー
- 4.3.2 要件定義における未決定事項の取扱い
- 4.3.3 要件定義文章の品質向上

※解説する部分

## 4.3.3 要件定義文章の品質向上

### ■ 目的

- 文章の曖昧さにより要件定義した内容がレビュー者や後工程の従事者に正しく伝わらないことを避ける

### ■ 留意点

- 市販のテキストなどにより、技術文書記載の技術を習得する
- 文書による冗長な説明に頼らず、なるべく**図や表を使用**した分かりやすい説明を用いて、理解しやすくする
- チェック強化・体制強化など**プロジェクト内での策を併用**する

※ 文書表現に問題が多く含まれる場合には、内容そのものにも誤りが含まれる場合が多い

## ■ 第5章で説明している事項

- 第5章では、第3章、第4章で触れた課題に関して、企業で実際に行っている対策事例を解説する。

## ■ 目次（関連部分）

5.1 「ビジネスに貢献する要求を見極める」の事例（3.1節） ※解説する部分

5.2 「要件量を可視化する」の事例（3.2節）

5.3 「業務バリエーションの整理」の事例（3.3節）

5.4 「非機能要求の進め方」の事例（3.4節）

5.5 「ステークホルダ分析」の事例（3.5節）

5.6 「要求の定量化による合意形成と膨らむ要求の制御」の事例  
（3.2節、3.4節、3.5節、3.6節）

5.7 「手戻りコストを抑制する要件定義書のレビュー」の事例（4.3.1項）

5.8 「要件定義文章の品質向上」の事例（4.3.3項） ※解説する部分

## 5.8 「要件定義文章の品質向上」 の事例

～図表を使った記述技法～

株式会社NTTデータ

## 4.3.3 要件定義文章の品質向上

日本語文章の曖昧さを低減しよう。「形容詞や助詞の使い方など」「表にできるものは表にする」「図やイラストで視覚化して伝える」がある。

## ビジネスルールの記法の使い分けと記述方法の明確化

株式会社NTTデータの取り組みを紹介する。

### 【取り組みの背景】

要件定義成果物の標準的な技法（UML,ER図など）があるが、ビジネスルール（ユーザが業務を遂行する上での決まりごと）に関しては、定着していない。多くは自然言語で記述され、曖昧、個人差のばらつきが発生しやすい。

分類	定義	使われる表現	使われる表現
『事実』	業務に関する真実を単に記述したもの。多くの場合、重要な業務用語間の結合や関係を表す。	<ul style="list-style-type: none"> <li>・「～は…である」</li> <li>・「～には…が存在する」</li> <li>・「～は…によって構成される」</li> </ul>	<ul style="list-style-type: none"> <li>・PCには[稼働中]、[休止中]、[修理中]の状態区分（ステータス）が存在する。</li> <li>・車には[普通]、[中型]、[大型]のサイズ（サブタイプ）が存在する。</li> </ul>
『契機』	特定の状況下で何らかのアクションを引き起こす。	<ul style="list-style-type: none"> <li>・「～の場合…をする」</li> <li>・「～であれば…をする」</li> </ul>	<ul style="list-style-type: none"> <li>・顧客区分が[プラチナ顧客]であれば、割引券を送付し、[普通顧客]であれば、セール案内状を送付する。</li> </ul>
『制約』	システムやそのユーザーが実行するかもしれないアクションを制限する。	<ul style="list-style-type: none"> <li>・「～だけが…できる」</li> <li>・「～の場合のみ…をする」</li> <li>・「もし～であれば…ができる」</li> </ul>	<ul style="list-style-type: none"> <li>・PCの状態区分が[休止中]の場合、利用申請ができる。</li> </ul>
『推論』	特定の条件が真のときに新しい知識を発見する。	<ul style="list-style-type: none"> <li>・「～の場合…とみなす」</li> <li>・「～によって…を判定する」</li> </ul>	<ul style="list-style-type: none"> <li>・車載総重量が[5000kg未満]の場合、サイズを[普通]とみなし、[5000kg以上11000kg未満]の場合、[中型]とみなし、[11000kg以上]の場合、[大型]とみなす。</li> </ul>
『計算』	特定の数式やアルゴリズムを使用して実行する計算。通常、記述されたままでソフトウェア要求となる。	<ul style="list-style-type: none"> <li>・「～から…を算出する」</li> <li>・「～の場合…（計算式）で算出する」</li> </ul>	<ul style="list-style-type: none"> <li>・PCの状態区分が[稼働中]のもの合計数を、マシンスペックごとに算出する。</li> </ul>

分類	記述内容	記法
『事実』	データモデル	ER図
	用語の定義	表（用語定義）
	項目の列挙	箇条書き
	分類の指定	表（分類）
	項目が存在する条件の指定	表（項目の存在条件）
	複数の値に対する関係の指定	計算式
	文字（記号、符号）列のフォーマットの定義	図（フォーマット） 表（フォーマットの適用）
	値の有効範囲の指定	数直線グラフ 表（値のセット）
	条件による事物などの定義	表（事物の成立条件）
	事物などの関係（関連と多重度）の定義	階層図、クラス図
行為に対する責任の割り当て	責任分担表	
『契機』	アクター毎のアクティビティの定義	アクティビティ図、BPMN
	事物などに対するアクター毎の操作の制限	CRUD図
『制約』	条件による行為の制限	デシジョンテーブル
		ステートマシン図
	事前に実施する行為の指定	表（状態毎の行為の可否） 表（対象毎の行為の可否）
	期間や期限の指定	アクティビティ図、BPMN
	値の増減に関する制限	期限チャート
『推論』	空間的な制約の指定	折れ線グラフ
	事物などの状態の変化の指定	空間配置図
	時間帯の指定	ステートマシン図
『計算』	条件の組み合わせによる出力パターンの指定	時間帯グラフ
	計算式の実行順序の指定	例示
	四則演算及び切り上げ切り捨て	デシジョンテーブル
	入力値によって変化する計算式の指定	計算式 階段グラフ 計算式 デシジョンテーブル

このような事を記述するなら

これを使いなさい

- 表
- 数直線グラフ
- **時間帯グラフ**
- **階段グラフ**
- 折れ線グラフ
- **階層図**
- **クラス図**
- **アクティビティ図**
- **責任分担表**
- **デシジョンテーブル**
- **ステートマシン図**
- **期限チャート**
- **空間配置図**
- **階段グラフ**
- など

自然言語を極力減らす

# 事物などの関係（関連と多重度）の定義

## 概要

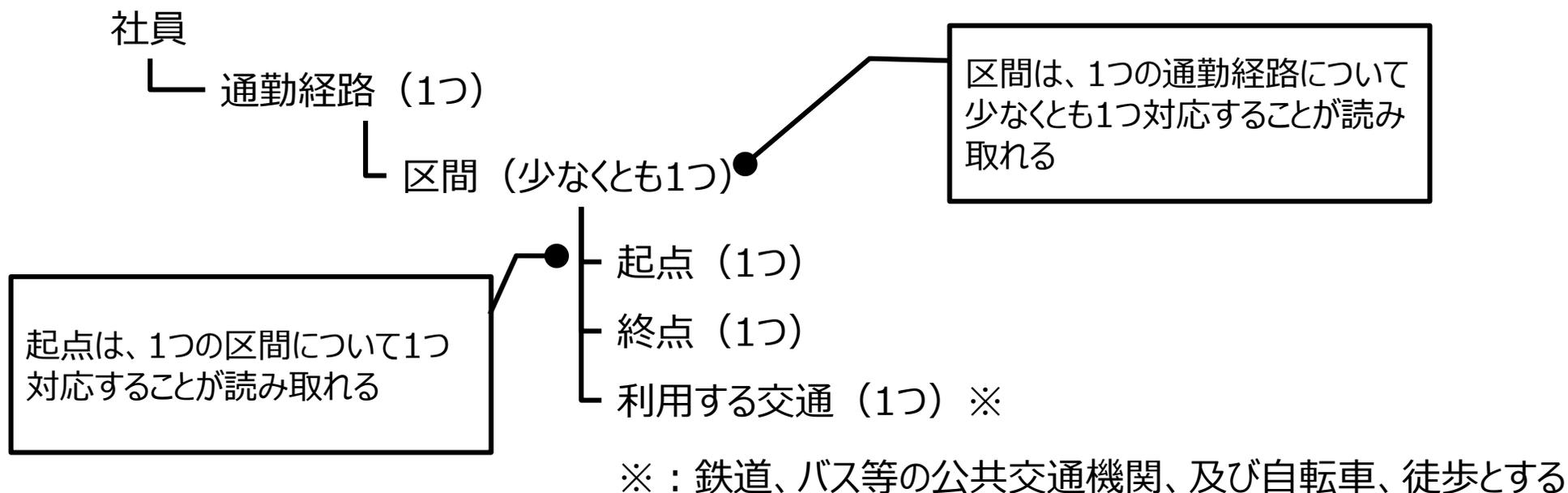
事物などを構成する項目間の関係（関連と多重度）が定義されている場合、その関係を階層図で記述する。

[Before]

社員は通勤費の支給対象となる通勤経路を1つ指定する。  
通勤経路は少なくとも1つの区間を指定する。1つの区間は起点、終点、利用する交通（鉄道、バス等の公共交通機関、及び自転車、徒歩）を各1つ指定する。

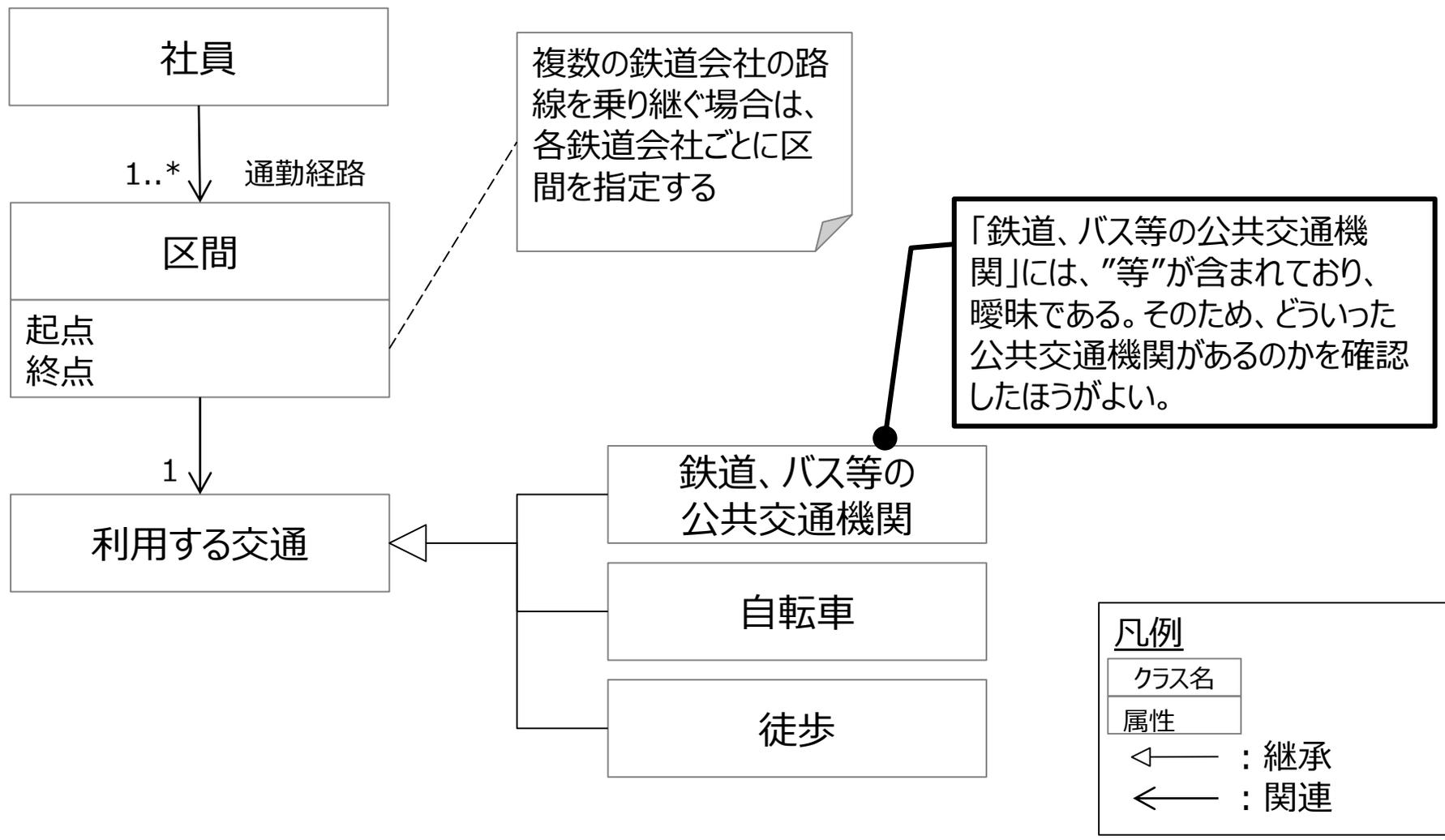
## 事物などの関係（関連と多重度）の定義（階層図）

[After]



事物などの構成項目間の関係を階層図として示すことにより、構成項目間の関係や、関連付けられる項目数が読み取りやすい

# 事物などの関係（関連と多重度）の定義（クラス図を用いた表現）



# 行為に対する責任の割り当ての定義

## 概要

行為毎にアクターの責任が指定されている場合、行為に対するアクターの責任を表で記述する。

[Before]

### 受注者となる共同企業体と発注者である機構の責任分担

- 事業提案書は受付期間内に必着とし、発送後であっても未着の場合の責任は、共同企業体に属するものとします。
- 共同企業体の提案内容に含まれる特許権等の法令に基づき保護される第三者の権利の対象となっている意匠、デザイン等を使用した結果生じる責任は、共同企業体が負うものとします。
- 機構は、自らの責任で契約期間内における設備の運用計画の変更を共同企業体に通知するものとします。

# 行為に対する責任の割り当ての定義（責任分担表）

[After]

受注者となる共同企業体と発注者である機構の責任分担は、以下の表の通り。

活動	責任	
	機構	共同企業体
事業提案書の受付期間内の必着		○
事業提案書が発送後であっても未着であること		○
共同企業体の提案内容に含まれる意匠、デザイン等（※）の使用		○
契約期間内における設備の運用計画の変更通知	○	

行為（活動）等をアクター（機構、共同企業体）の実行責任が明示できるため、何がどの行為に責任を負うかを明確に読み取れる

責任分担表では、通常役割（責任、管理、承認、説明など）を示す記号が使われるが、この例のように責任のみを示す場合は「○」と記載することで、表が読み取りやすくなる

※「意匠、デザイン等」は、特許権等の法令に基づいて保護され第三者の権利対象になるものを指している

どのアクターがどの行為に対して責任があるかを把握しやすい

# 行為に対する責任の割り当ての定義（RACIチャートを用いた表現）

活動	役割分担			
	ユーザ			ベンダ
	業務部門	システム部門	運用部門	
ビジネスプロセス関連図	RA	C	CI	C
業務機能構成図	RA	C	CI	C
画面／帳票レイアウト	A	R	CI	C
業務処理定義書	RA	C	CI	C
概念データモデル	A	R	CI	C
エンティティ定義書／データ項目定義書	A	R	CI	C
～中略～				
総合テスト計画書	RA	R	R	C
～中略～				
非機能要件書	A	R	C	C

**凡例**  
 R(Responsible): 実行責任者  
 A(Accountable): 説明責任者  
 C(Consulted): 協議先  
 I(Informed): 報告先

# アクター毎のアクティビティの定義

## 概要

アクターに対して、行為が割り当てられている場合、アクター毎に行為を図示する。

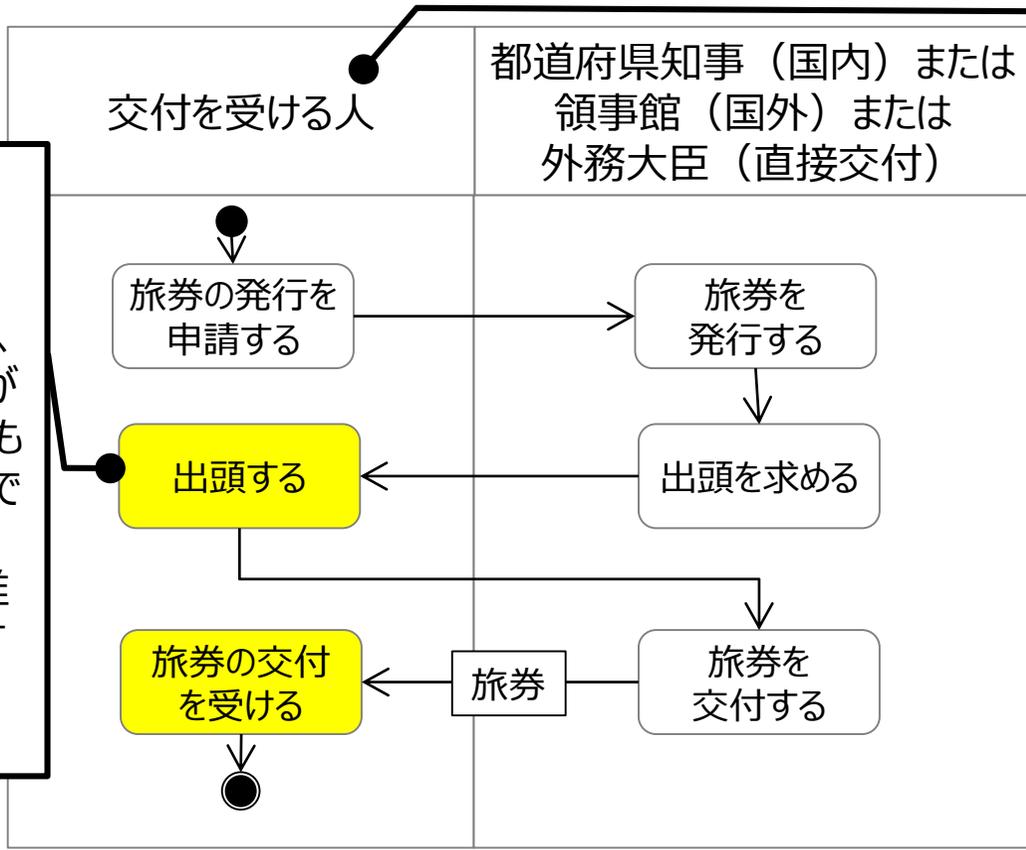
[Before]

規定により発行された一般旅券は、国内においては都道府県知事が、国外においては領事官が、外務省令で定めるところにより、当該一般旅券の発給につき申請をした者の出頭を求めて当該申請者に交付する。ただし、規定により直接外務大臣に申請する場合には、外務大臣が当該申請をした者の出頭を求めて当該申請者に交付する。

# アクター毎のアクティビティの定義（アクティビティ図）

[After]

規定により発行された一般旅券は、外務省令で定めるところにより、以下の頭の通り交付される。



交付を受ける人のレーンにより、交付を受ける人自身が実施すべき行為を容易に読み取ることができる

「出頭する」「旅券の交付を受ける」の行為は、ルールに明示的に書かれていないが、行為の対応関係から推測が付く場合は、推測した行為も記載する。この例では黄色で色付けしている。これにより、レビューの際に推測が正しいかどうかを確認することができる

**凡例**

- : 開始
- : 終了
- : アクティビティ
- : 遷移

アクティビティ図のレーンを見ることで、同一人物に禁止されている行為や、同一人物がすべき行為を読み取りやすい。

# 条件による行為の制限

## 概要

条件によって行為が制限される場合、表、デシジョンテーブル、ステートマシン図で記述する。

デシジョンテーブル	: 条件の組み合わせによって、行為を制限する場合
ステートマシン図	: 状態によって、次に実施する行為を制限する場合
表（状態ごとの行為の可否）	: 状態ごとに行為の可否を制限する場合
表（対象ごとの行為の可否）	: 行為の対象によって、アクターの行為の可否を制限する場合

## [Before]

### ①受験者の確認

試験会場（研修室）入り口で受験者持参の「写真付き証明書」で本人確認をする。

- ・「写真付き証明書」がない ⇒ 社員証、健康保険証等で本人確認ができれば例外的に受験可能。
- ・本人でない ⇒ 受験不可。
- ・受験日が違った ⇒ 受験不可。
- ・「受験票」がない ⇒ 受験票のコピーと照らし合わせ本人確認ができれば受験可能。

### ②受験者への説明

受験者に受験票を返し、オリエンテーション開始までに必ず着席するよう伝える。また、試験中はトイレ休憩がないため、事前に行くようあわせて伝えること。

### ③遅刻者の対応

試験受付時間内に受付をしない場合、受験することができない。但し、交通機関の遅延に限り、遅延証明書があり試験開始時刻を過ぎてない場合のみ受験を認める。

- ・交通機関の遅延（遅延証明書がない）⇒ 受験不可
- ・その他の理由による遅刻（体調不良等）⇒ 受験不可
- ・交通機関の遅延（試験開始時刻を過ぎた）⇒ 受験不可
- ・交通機関の遅延（遅延証明書がある、試験開始時刻を過ぎていない）⇒ 受験可能

# 条件による行為の制限 (デシジョンテーブル)

			1	2	3	4	5	6	7	8
条件	写真付き証明書の有無	あり	Y	-	Y	-	-	-	-	-
		なし	-	Y	-	Y	-	-	Y	-
	写真付き証明書で本人確認	本人である	Y	-	Y	-	-	-	-	-
		本人でない	-	-	-	-	Y	-	-	-
	受験日	合っている	Y	Y	Y	Y	-	-	-	-
		違っている	-	-	-	-	-	Y	-	-
	受験票の有無	あり	Y	Y	-	-	-	-	-	-
		なし	-	-	Y	Y	-	-	-	Y
	社員証、健康保険証等で本人確認	一致	-	Y	-	Y	-	-	-	-
		不一致	-	-	-	-	-	-	Y	-
	受験票のコピーと照らし合わせ本人確認	一致	-	-	Y	Y	-	-	-	-
		不一致	-	-	-	-	-	-	-	Y
動作	受験可能		X	X	X	X	-	-	-	-
	受験不可		-	-	-	-	X	X	X	X

**凡例**  
**【条件】**  
 Y:条件が真  
 -:条件に無関係  
**【動作】**  
 X:動作が生じる  
 -:動作が生じない

可読性向上  
 条件の組み合わせの漏れ、誤り低減

# 条件による行為の制限 (デシジョンテーブルの記述ポイント)

## ■ 不完全なデシジョンテーブル

ビジネスルール① 「A」の場合「B」とみなす  
年「10万円以上」の購入の場合を「優良」とみなす。  
年「10万円未満」の購入の場合を「普通」とみなす。

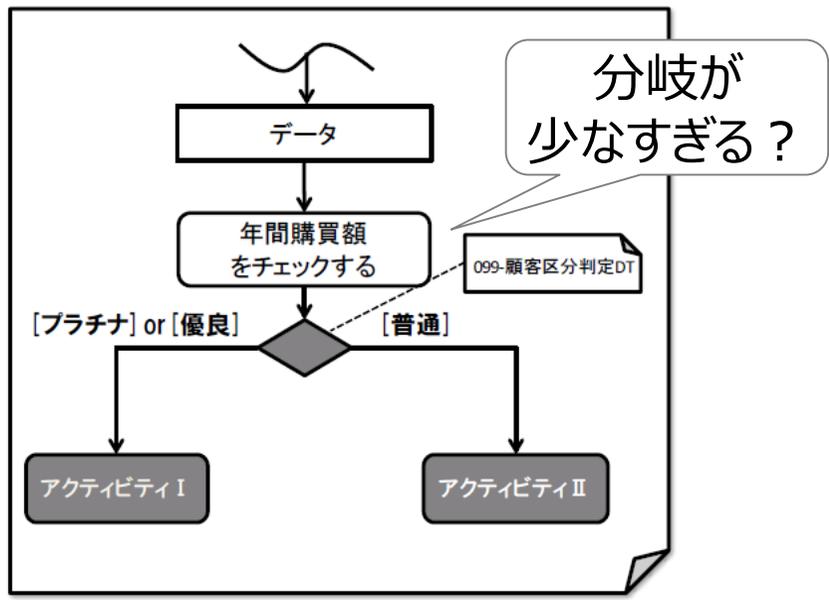
ビジネスルール② 「X」によって「Y」を判定する  
「年間購買額」によって「顧客区分」を判定する。

条件部	判定部
年間購買額	顧客区分
10万円以上	優良
10万円未満	普通

## ■ デシジョンテーブルとアクティビティ図の不整合

条件部	判定部
年間購買額	顧客区分
50万円以上	プラチナ
10万円以上 50万円未満	優良
1円以上 10万円未満	普通
0円	新規

バリエーションが多すぎる？



# 期間や期限の指定

## 概要

ある事由に対する実施内容とその期限が指定されている場合、事由と実施内容と期限の時系列関係を図示し、さらに、具体的日付を入れた例も合わせて記述する。

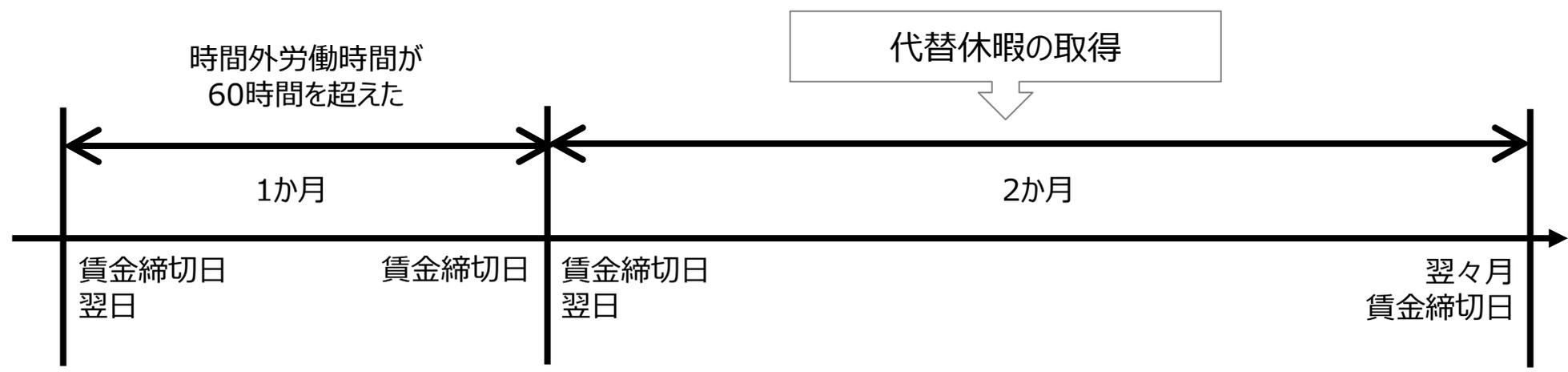
[Before]

- 第1項 賃金締切日から翌賃金締切日までの1か月の時間外労働が60時間を超えた労働者に対して、労使協定に基づき、次により代替休暇を与えるものとする。
- 第2項 代替休暇を取得できる期間は、賃金締切日翌日から翌賃金締切日までの1か月の時間外労働が60時間を超えた翌日から起算して、翌々月の賃金締切日までとする。

# 期間や期限の指定（期限チャート）

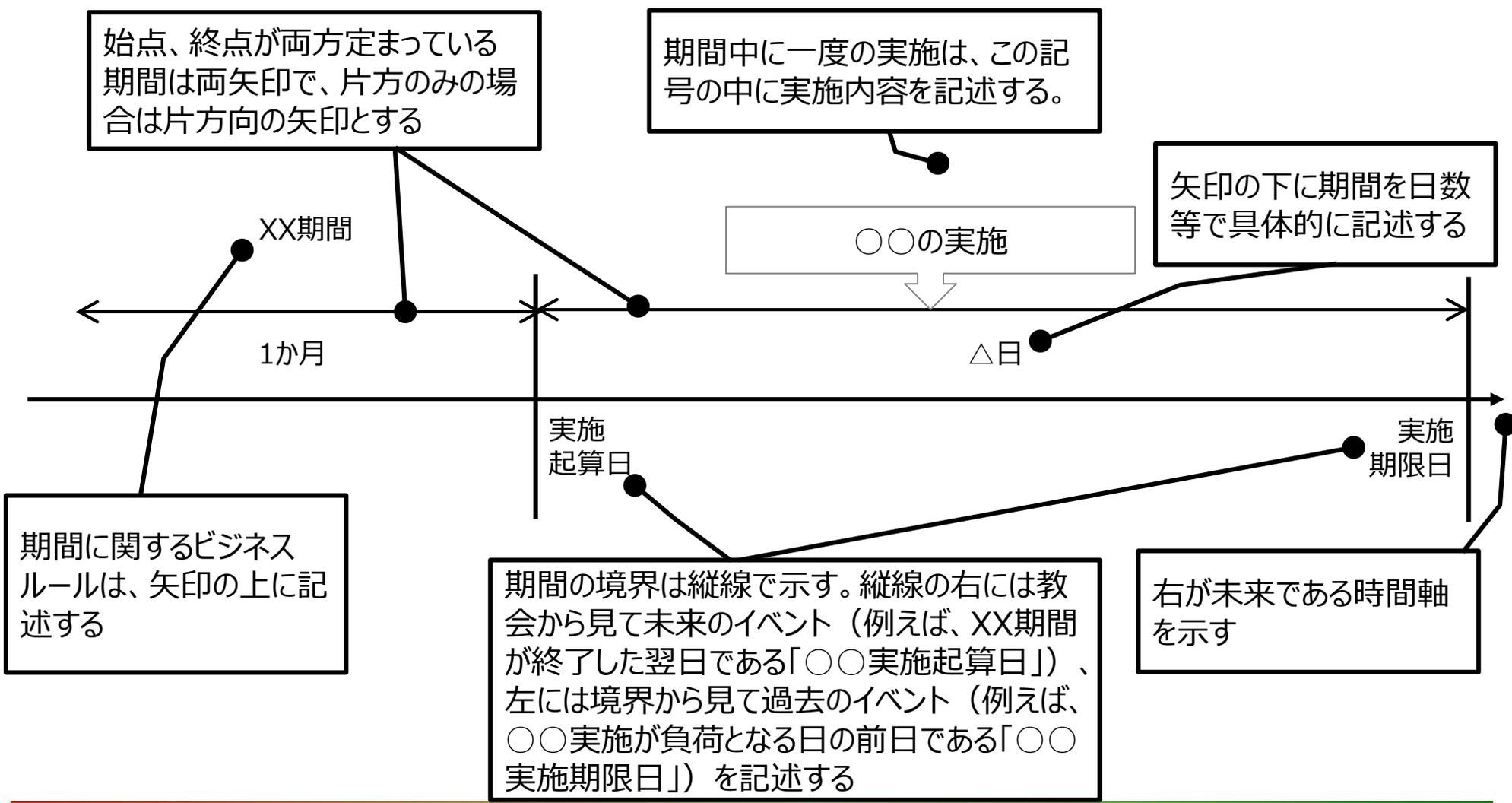
[After]

1か月の時間外労働が60時間を超えた労働者に対して代替休暇を与える。  
 ここで1ヶ月は賃金締切日翌日から翌賃金締切日までとする。  
 代替休暇を取得できる期間は以下とする。



条件に対して、実施する行為の期限が、時間の長さとして視覚的に理解しやすい

# 期間や期限の指定 (期限チャートの記述ポイント)



# 空間的な制約の指定

## 概要

上下、左右、前後などの位置関係が制約条件となる場合、位置関係を図示する。

[Before]

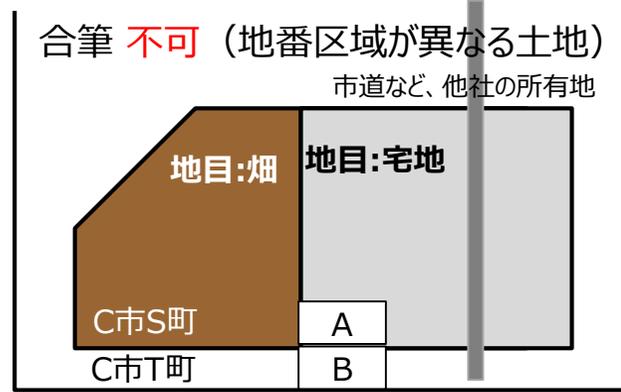
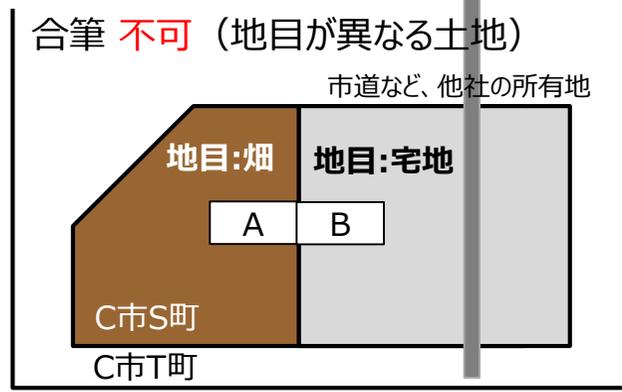
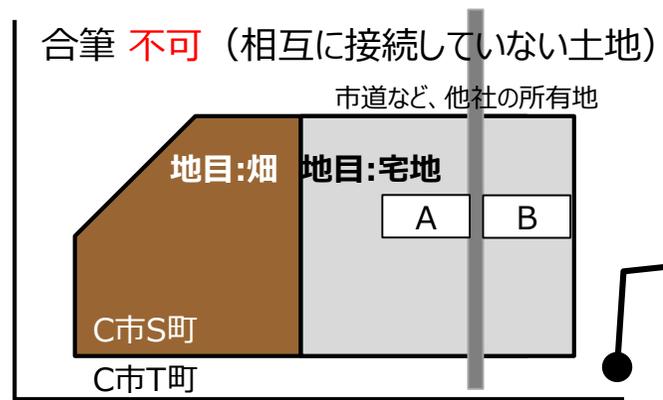
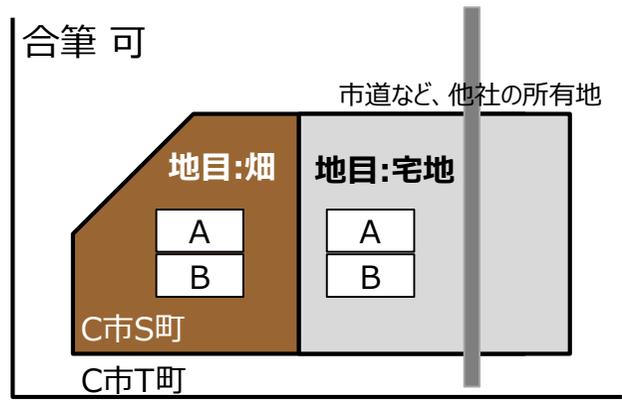
次に掲げる合筆の登記は、することができない。

- 一、 相互に接続していない土地の合筆の登記
- 二、 地目又は地番区域が相互に異なる土地の合筆の登記

# 空間的な制約の指定 (空間配置図)

[After]

合筆の可否は、以下の図の通り



土地の配置を図示することにより、土地の合筆条件（土地が隣接する様子）を視覚的に把握しやすい。

上下、左右、前後などの位置関係を図で示すことにより、空間的な制約を視覚的に理解しやすい。

# 事物などの状態の変化の指定

## 概要

事物などの状態の変化が指定されている場合、その状態の変化と状態が変化する時の条件を図示する。

[Before]

第1項 労働者が、次のいずれかに該当するときは、所定の期間**休職**とする。

- ① **業務外の傷病による欠勤**が3か月を超え、なお療養を継続する必要があるため勤務できないとき（2年以内）
- ② **業務を理由とする傷病により休職**させることが適当と認められるとき（必要な期間）

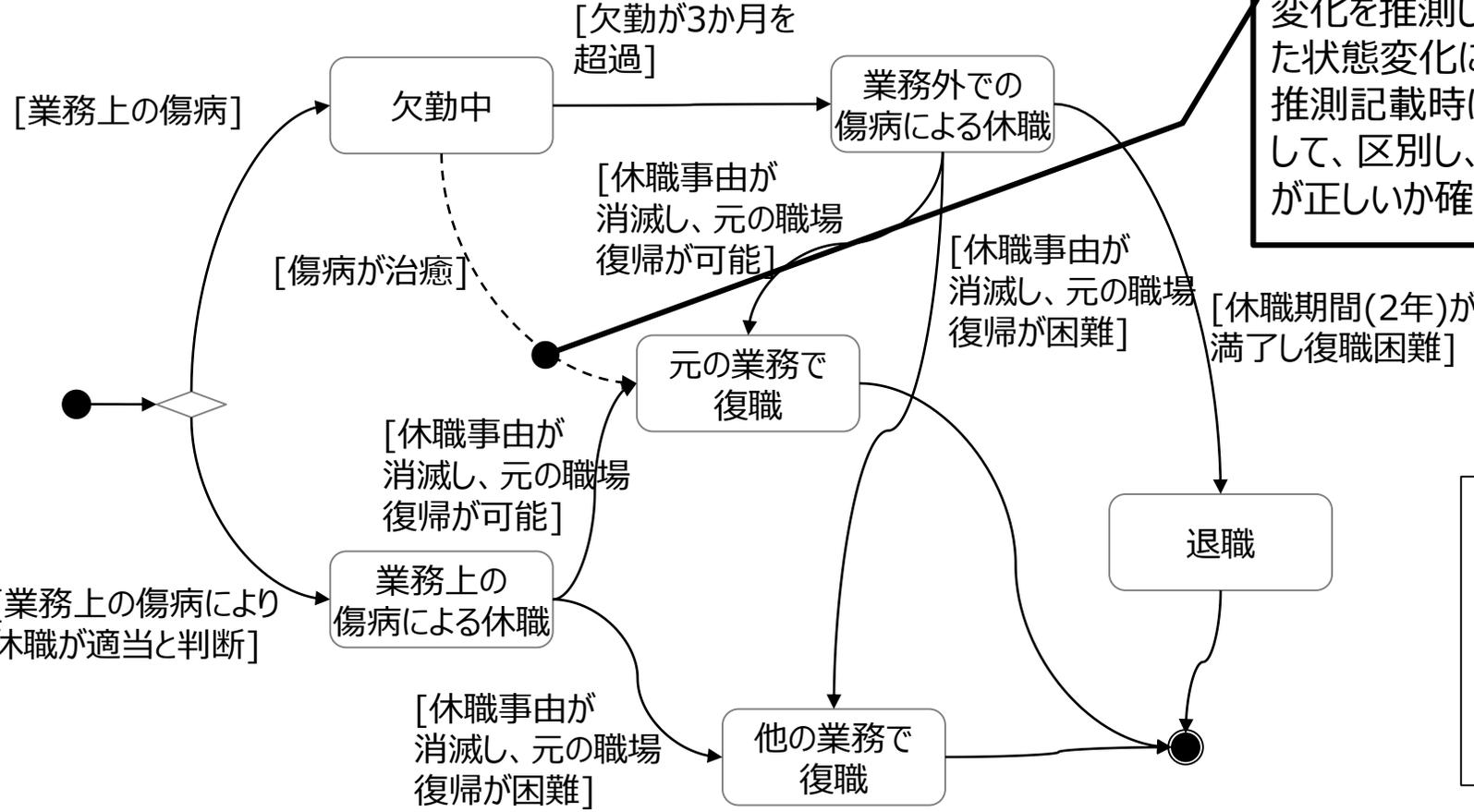
第2項 休職期間中に休職事由が消滅したときは、原則として**元の職務に復帰**させる。ただし、元の職務に復帰させることが困難又は不適當な場合には、**他の職務に就かせる**ことがある。

第3項 第1項第1号により休職し、休職期間が満了してもなお傷病が治癒せず就業が困難な場合は、休職期間の満了をもって**退職**とする。

# 事物などの状態の変化の指定 (ステートマシン図)

[After]

労働者の休職から職務への復帰または退職を以下のように定める。



ルールに記載されていない状態の変化を推測した場合は、推測した状態変化についても記載する。推測記載時には点線を使うなどして、区別し、レビュー時に推測が正しいか確認することができる

**凡例**

- : 開始
- : 終了
- : 状態
- : 遷移
- ⤵ : 分岐

どの状態がどのような条件で変化するかが読み取りやすい。  
 状態が変化する条件を確認すれば、条件分岐の誤りや漏れを発見できる。

# 時間帯の指定

## 概要

時間帯が指定されている場合、実施可能または実施禁止である時間帯を図示する。

[Before]

使用者は、満18歳に満たない者を午後10時から午前5時までの間において使用してはならない。ただし、交替制によって使用する満16歳以上の男性については、この限りでない。

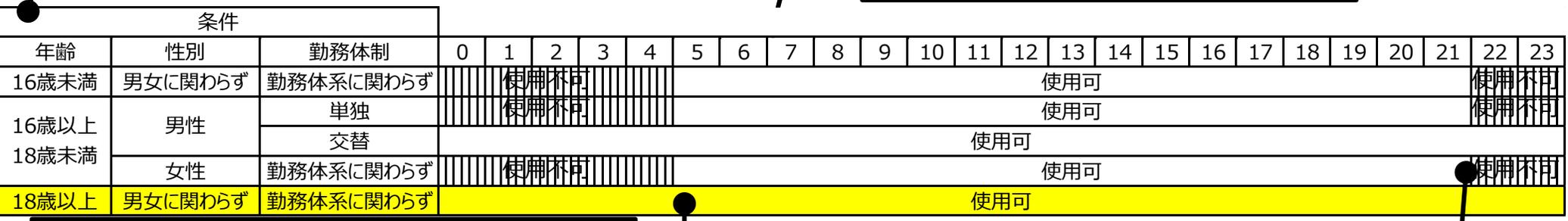
# 時間帯の指定（時間帯グラフ）

[After]

使用者が、労働者を使用可能な時間、使用不可能な時間は、以下の通りとする。

左側に条件を網羅するように記載する。例えば、（男性、女性）の組、または（不問）となるように記載する。これにより、条件の組み合わせの網羅性をチェックできる

1日の中の時間帯を示すので、24時間の帯グラフの枠を用意する。これにより、24時間における使用可否が視覚的に読み取れる



推測した時間帯の指定も記載する。その際には色付けするなどして、推測した時間帯の指定であることを区別できるようにする。これによりレビューの際に推測が正しいかを確認できる

使用不可の時間帯が何時から何時までであるかが、直感的に見ることができる

対象毎に実施可能または実施禁止の時間帯を、長さとして視覚的に理解しやすい。  
対象毎に時間帯の重なりを見ることで、未定義箇所を発見できる。  
変更可能範囲を例で示すことによって、どう変更可能かがわかりやすい。



# 入力値によって変化する計算式の指定

## 概要

入力値によって変化する計算式が指定されている場合、入力値と変化する計算内容を表やグラフで記述する。

[Before]

大人片道普通旅客運賃は、次の各号に基づき営業キロを運賃キロに換算したものを適用する。

(1) 営業キロ1キロメートル以上100キロメートル以下

営業キロ5キロメートル以下は1キロごとに分割し、1キロメートル以上5キロメートル以下は営業キロと等しくし、営業キロ6キロメートル以上は、5キロごとに分割し、1分割単位を増すごとに5キロメートルを加算したものを運賃キロとする。

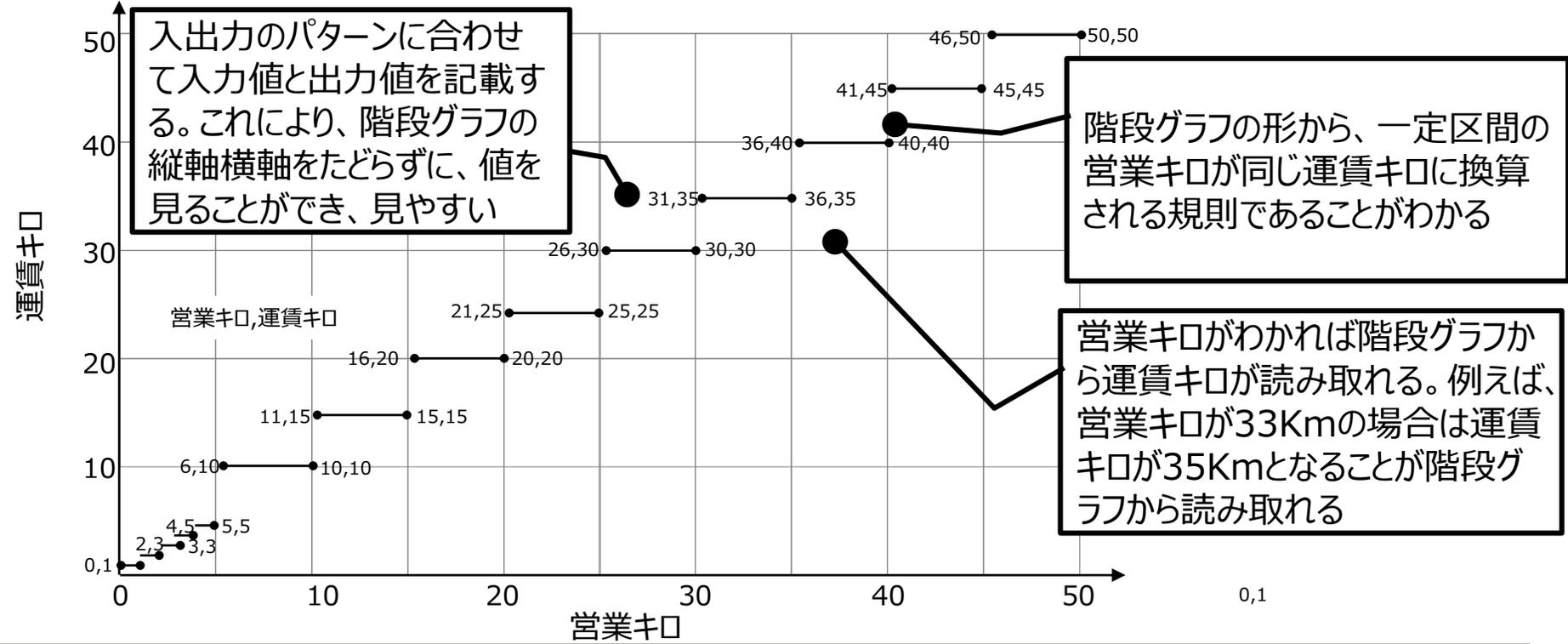
※営業キロ101キロメートル以上は割愛

# 入力値によって変化する計算式の指定 (階段グラフ)

[After]

大人片道旅客運賃は、営業キロ（1キロメートル単位）を運賃キロに換算したものを適用する。  
営業キロから運賃キロへの換算方法を以下に示す。

- (1) 営業キロ5キロメートルごとに分割する。
- (2) 1キロメートル以上5キロメートル以下は、運賃キロを営業キロと等しくする。
- (3) 6キロメートル以上は、1分割単位を増すごとに5キロメートルを加算したものを運賃キロとする。



条件分岐に応じて、計算内容がどのように決められているかが、わかりやすい。  
条件分岐を順に見れば誤りや漏れを発見できる。  
グラフを見れば、入力に対する出力値がすぐにわかる。  
入力に対して出力値が漏れなく定義されているか、グラフをかけないパターンがないかを確認すれば、誤りや漏れを発見できる。

## 最後に

本事例では、自然言語で表現しがちなビジネスルールを図や表を用いて表現する技法をまとめ、要件定義書を書く際のガイドラインとして公開し実践しています。

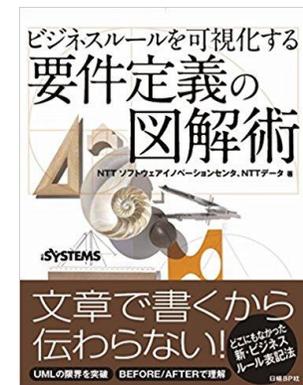
本技法を用いて表現することにより、要求の抽出抜け漏れを防ぐだけでなく、可読性の向上にも寄与します。

可読性向上のメリットは、要求のレビューアがレビューを質良く、効率的に実施できるだけでなく、後続工程において、開発する開発者へ要求の内容を抜け漏れ、あいまい性無く伝達できるメリットもあります。

### □ (ちょこっと宣伝)

本技法は、書籍「**ビジネスルールを可視化する 要件定義の図解術**」としてご覧いただけます

ビジネスルールを可視化する 要件定義の図解術  
NTTソフトウェアイノベーションセンタ、NTTデータ(著)  
日経BP社発行



ご清聴ありがとうございました

**SEC BOOKS : ユーザのための要件定義ガイド**

～要求を明確にするための勘どころ～

<http://www.ipa.go.jp/sec/publish/tn16-008.html>

**システム構築の要件定義に役立つポイント集を公開**

～家づくりを例に、要求を明確にする勘どころ概説～

<https://www.ipa.go.jp/sec/reports/20180327.html>

# (付録)

## ■ 4.2 主要な成果物と作成上の留意点

- 4.2.1 ビジネスプロセス関連図
- 4.2.2 業務機能構成表
- 4.2.3 ビジネスプロセスフロー（業務フロー／システム化業務フロー）
  - コラム 新しいビジネスプロセスを作り上げる意識を持つ
- 4.2.4 画面／帳票レイアウト
- 4.2.5 業務処理定義書
- 4.2.6 概念データモデル（ER図）
  - コラム 概念データモデル（ER図）を業務部門に確認する
- 4.2.7 エンティティ定義書／データ項目定義書
- 4.2.8 CRUD図
- 4.2.9 総合テスト計画書
- 4.2.10 システム移行計画書
- 4.2.11 運用・操作要件書
- 4.2.12 非機能要件書

## 4.2.1 ビジネスプロセス関連図（1）

### ■ 目的

- To-Beのビジネスプロセスを検討する。共有する。
- 開発対象の範囲と他プロセスとの関連を明確にする

### ■ 作成のポイント

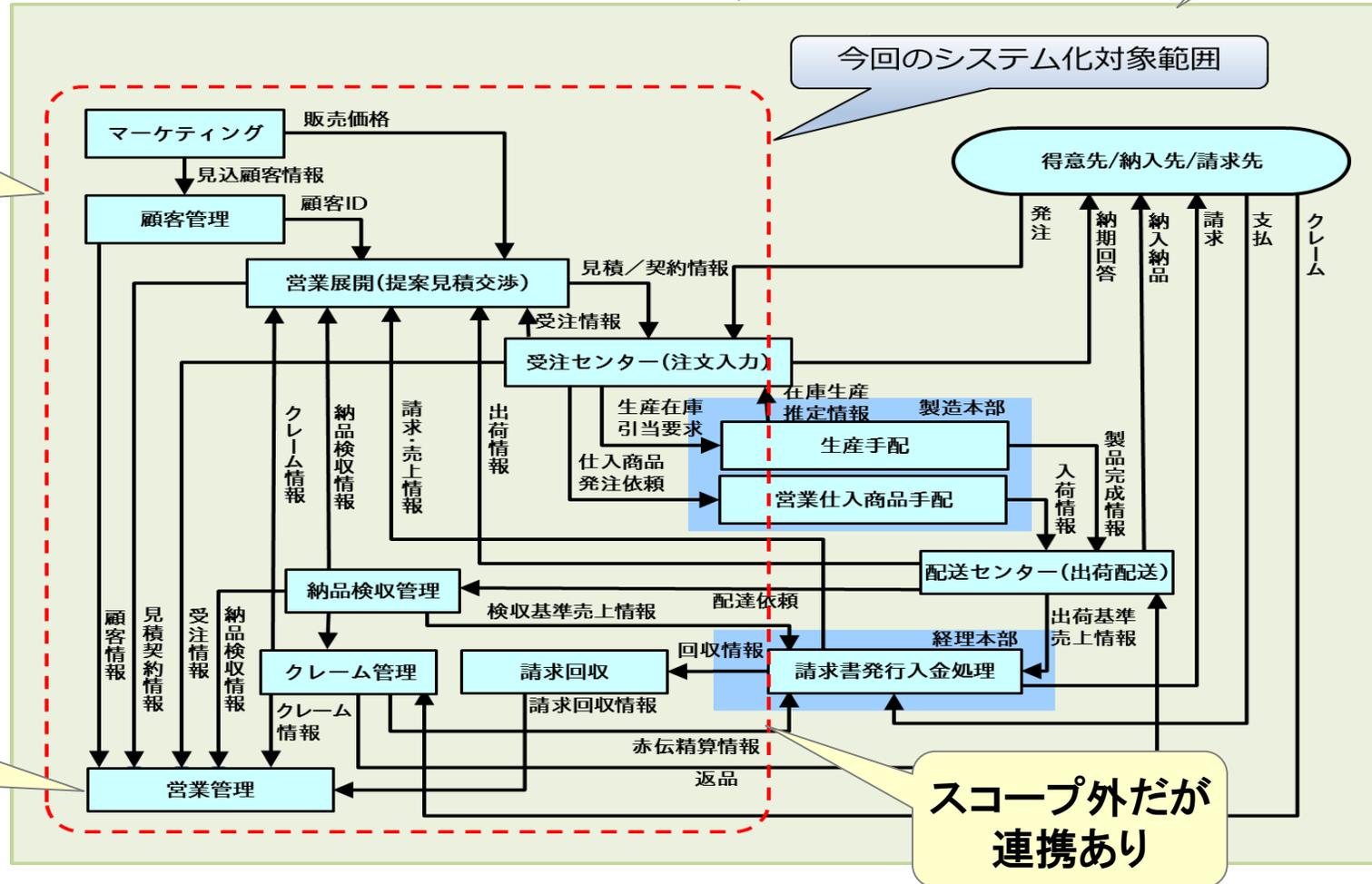
- **最初**に作成する
- 全体像を鳥瞰できるように **1枚に表す**
- 境界線の内と外の区別が明確にわかるようにする
- 全体最適を視野に入れてTo-Beを描く  
既存業務の変更を伴う場合は、As-IsとTo-Beの2種類を作成する
- プロセス（組織・機能）間の情報連携の発生元・連携先・用途を、システム化対象部門（事業部門など）だけでなく、関連部門（経理・人事・総務・監査・購買・物流など）からヒアリングする

# 4.2.1 ビジネスプロセス関連図 (2)

## ■ ビジネスプロセス関連図の例

最初に作成して意識を合わせる

1枚で表す



## 4.2.2 業務機能構成表（1）

---

### ■ 目的

- 対象範囲の業務を構成する業務機能を洗い出す
- 各業務に含まれる機能をシステム化対象にするかどうかを区分する

### ■ 作成のポイント

- 対象システムの業務機能を担当者の作業レベルまで分解する
- **3階層**（大分類、中分類、小分類）に機能を分けるのが適切
- システム化対象を抽出し、優先度を設定する

# 4.2.2 業務機能構成表 (2)

## ■ 業務機能構成表の例

業務レベル1	業務レベル2	業務レベル3	スキル	担当者	作業時間 (分/月)		システム化対象	重要度	優先順位	
					現在	3年後				
図書管理	図書購入	図書購入依頼、受取	A	田中	120	同左	人手	-	-	
		新規登録	B	田中	60	同左	○	1	1	
		管理	B	田中	120	同左	○	2	2	
		貸出期間	B	田中	1,200	同左	○	2	2	
			図書在庫照会への回答など	B	田中	60	同左	○	1	1
			会員登録	B	田中	60	同左	○	1	1
有料会員管理		変更、継続	B	田中	60	同左	○	1	1	
		退会	B	山田	600	同左	○	1	1	
		請求	B	山田	480	同左	○	1	1	
		入金確認	B	山田	420	同左	人手	-	-	
貸出管理		督促	B	山田	840	同左	人手	-	-	
	貸出	新規貸出	C	鈴木	8,400	同左	○	1	1	
		ネット貸出	B	中村	2,100	4,200	○	2	2	
ネット照会、予約管理	返却	返却、督促など	B	鈴木	2,400	同左	人手	-	-	
	期間延長	期間延長受付	B	鈴木	1,200	同左	人手	-	-	
	ネット照会	棚検索、貸出可能照会	C	中村	2,400	4,800	○	3	3	
ネット予約	ネット予約	予約受付、引当通知	B	中村	4,800	9,400	○	3	3	
		督促	B	中村	4,800	9,400	○	3	3	

業務プロセスを階層的に洗い出す

システム化対象でないものも含める

システム化の対象となるものにマークする

必要スキル A：熟練を要する B：AとCの中間 C：短時間で習得できる  
 重要度 1：最重要 2：次に重要 3：重要度低い  
 優先順位 1：一次 2：できれば一次、二次でも可 3：二次以降で問題ない

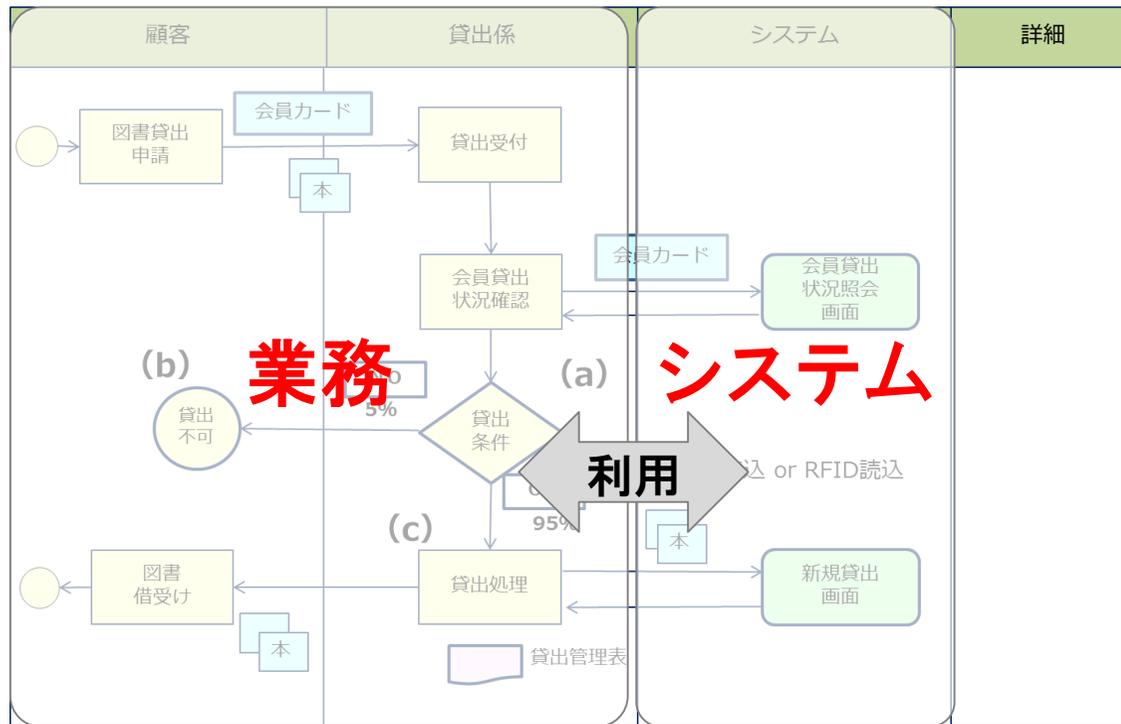
### ■ 目的

- As-Is : 業務の問題点や課題を共有する
- To-Be : 新しいビジネスプロセスを示して共有する

### ■ 作成のポイント

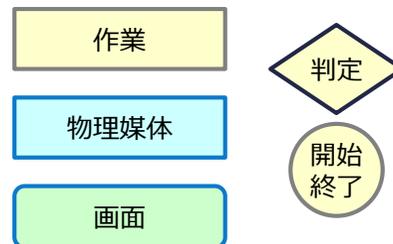
- 業務フロー
  - 手段に依存しない目的レベルの**業務を明確にする**
- システム化業務フロー
  - ITシステムを利用した作業の流れと、**システムとの関連について記載する**

## ■ ビジネスプロセスフローの例



- (a)
  - 条件によって処理が大きく変わるものだけを記載する
  - 分岐の一方は別のプロセスにして、できるだけ複雑な図になることを避ける
  - 判定条件は詳細欄に記載する  
詳細欄に記述しきれない量になる場合には、業務処理定義書に記載する
- (b)
  - 継続する業務プロセスがある場合は、業務フローを別に作成し、コネクタを記載する
- (c)
  - 条件の詳細は、詳細欄にコメントを記載する
  - 条件の組み合わせが二段階を超える場合は詳細欄に条件分岐内容を記述し、フローが複雑にならないようにする

最上部に担当部署（アクタ）およびシステムを記し、各担当部署に対応する業務の流れをその下に記している



# 抜け漏れ低減の観点

## ■ 5W2Hによる確認ポイント

5W2H	確認ポイント
What	・商品やサービス(What)による違いはないか
Who	・相手(Who)による違いはないか
When	・時間、時期、タイミング、順序など(When)による違いはないか ・状況や状態、条件による違いはないか
Where	・場所(Where)による違いはないか
Why	・理由(Why)による違いはないか
How	・やり方や方式(How)による違いはないか
How many (How much)	・量(How many)や金額(How Much)による違いはないか

**5W2H**の観点で展開した業務のバリエーションが、ビジネスプロセス上にすべて記載されているのかを確認することにより、**抜け漏れが低減**できる

## ■ 例外処理、特殊処理が漏れやすいので注意する

### ● 例外処理

- 業務プロセス上のある処理を実行した際に、処理を継続できないようなエラーが発生した際に実施する業務上の別処理

### ● 特殊処理

- イベント、キャンペーンなどの不定期・低頻度で実行される業務処理

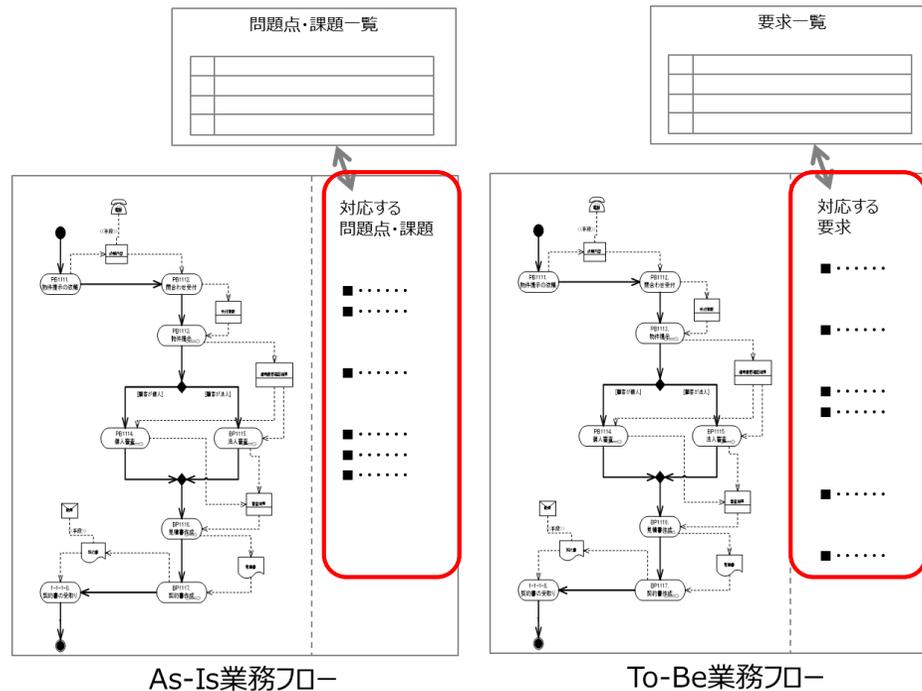
## ■ 抜け漏れ防止のポイント

- 入力ミス、決算またぎ、責任者不在、緊急時の事務、サービス時間外の対処など、重点的に確認すべきポイントを事前に洗い出す
- 以下の方法により抽出および漏れのチェックを行う
  - 業務担当者からのヒアリングの実施
  - 抽出結果に対する別担当者による再確認
  - 現場で実施されている操作の実見 など

# 【コラム】～新しいビジネスプロセスを作り上げる意識を持つ～

課題：ビジネスプロセスを見直したが、さほど変化しなかった

対策：フロー図に要求・課題の欄を設け、それらとマッピングさせながらフローの検討を進めること



プロセス改革の検討においては、より経営レベルに近いビジネスプロセスビジネスプロセスから検討を始める

## 4.2.4 画面／帳票レイアウト（1）

### ■ 画面レイアウト作成の目的

- 画面上に配置されたデータ項目（入力項目、表示項目）や機能（ボタンなど）が業務部門の要求と合っているかを確認する
- 画面操作方法、画面遷移が、業務部門の要求と合っているかを確認し、どのように入出力すると使いやすいかイメージを検討する

### ■ 画面レイアウト作成のポイント

- **まず標準化**をする
- 目的や使い方を明確にし、それに適した画面レイアウトになっているかを業務部門と確認する
- **設計工程との分担（詳細定義は設計にて行う場合）**  
新規：画面イメージ、改造：既存画面レイアウトに赤入れ  
プロトタイピング、モックアップを要件定義で実施することも多い  
見積もれることが重要

## 4.2.4 画面／帳票レイアウト（2）

### ■ 帳票レイアウト作成の目的

- 帳票の機能要件について、利用者と開発者の合意形成を図る
- 帳票レイアウトに記載された出力情報や業務フローなどをもとに、帳票がシステム化の目的達成に役立つかを確認する

### ■ 帳票レイアウト作成のポイント

- **まず標準化**をする
- 紙で出力するか、電子化するかを考える
- CSVダウンロード等の別の選択肢を取ることも検討する
- **設計工程との分担（詳細定義は設計にて行う場合）**  
新規：帳票イメージ、改造：既存帳票レイアウトに赤入れ  
見積もれることが重要

# 4.2.4 画面／帳票レイアウト (3)

## ■ 画面レイアウト例

使いやすさ、ユーザビリティへの考慮

モックアップやプロトタイプングなどによる確認

操作者が業務実行のために必要とする機能メニュー

新規、照会、変更、削除のうち必要事項を表示

クリックするとポップアップメニューで項目を選択

上段にHeader 下段にDetailを表示

エラーメッセージ等を表示

No.	更新	削除	* 製品コード	製品名	製品分類	* 受注数値
1	<input type="checkbox"/>	<input type="checkbox"/>	Rak.FW	楽天 Framework II	開発ツール	1
2	<input type="checkbox"/>	<input type="checkbox"/>	Rak.WF	楽天 Workflow II	改善管理	4
3	<input type="checkbox"/>	<input type="checkbox"/>	QSPortal	QuickSolution Portal	検索エンジン	1
4	<input type="checkbox"/>	<input type="checkbox"/>	Rak.FW	楽天 Framework II	開発ツール	4
5						
6						
7						
8						
9						
合計						10

## 4.2.5 業務処理定義書（1）

### ■ 目的

- 各業務処理の具体的な内容（業務ルール）を明確にする
  - To-Beの業務ルールを立案し、合意を形成する
  - 業務そのものを理解する（関係者と共有する）

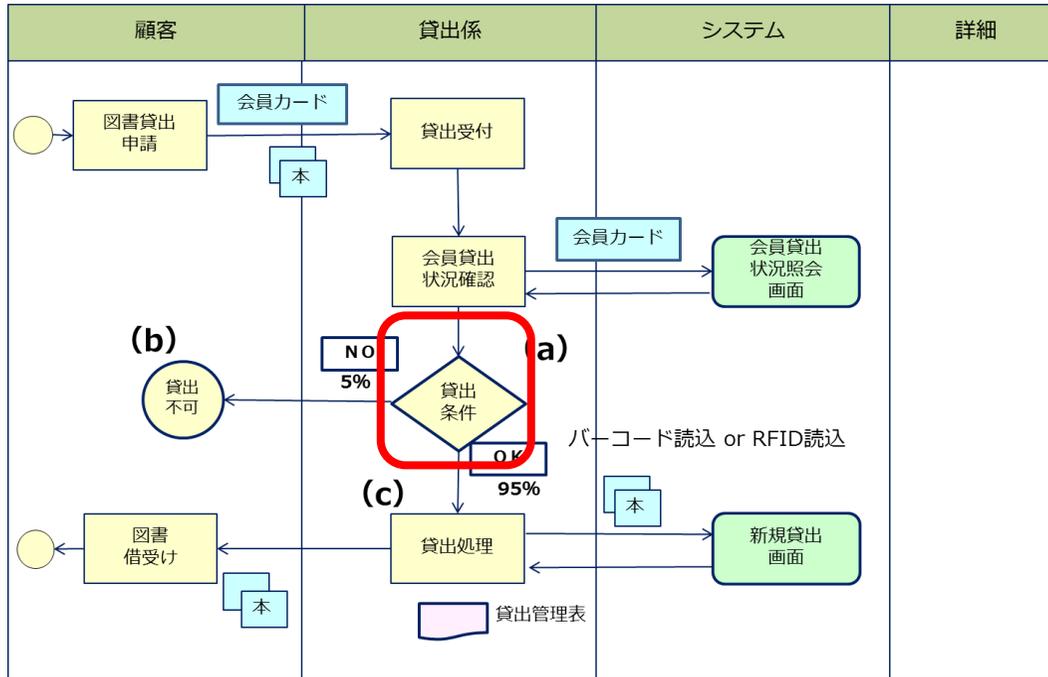
### ■ 作成のポイント

- 業務の背景、目的、取り扱い範囲・制約事項を明確にしておく
- **表形式**にまとめると漏れや不整合を発見しやすい
- 重要な時間や日付、発生・完了タイミングを確認する

# 4.2.5 業務処理定義書 (2)

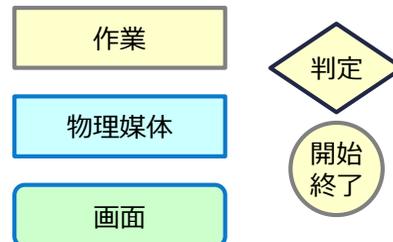
## ■ 業務処理定義書を作成する処理の例

ビジネスプロセスフローの例 再掲



- (a)
  - 条件によって処理が大きく変わるものだけを記載する
  - 分岐の一方は別のプロセスにして、できるだけ複雑な図になることを避ける
  - 判定条件は詳細欄に記載する  
詳細欄に記述しきれない量になる場合には、**業務処理定義書**に記載する
- (b)
  - 継続する業務プロセスがある場合は、業務フローを別に作成し、コネクタを記載する
- (c)
  - 条件の詳細は、詳細欄にコメントを記載する
  - 条件の組み合わせが二段階を超える場合は詳細欄に条件分岐内容を記述し、フローが複雑にならないようにする

最上部に担当部署（アクタ）およびシステムを記し、各担当部署に対応する業務の流れをその下に記している



# 4.2.5 業務処理定義書 (3)

## ■ 業務処理定義書の例

(判定条件1) (判定条件2)		(判定条件3)			
会員種別	会費 納入状況	判定項目	現在貸出中の図書の返却日延長回数		
			なし	1回	2回以上
無料会員	-	返却日 延長可否	延長可		延長不可
		追加貸出 可否	合計3冊まで 貸出可能	追加貸出 不可	
有料会員	-	返却日 延長可否	延長可	延長不可	
		追加貸出 可否	合計3冊まで 貸出可能	追加貸出 不可	
	会費 督促中	返却日 延長可否	延長可		延長不可
		追加貸出 可否	合計3冊まで 貸出可能	追加貸出 不可	

3パターン

3×4の12通りに展開して  
すべてのパターンの  
対応を表すことにより  
抜け漏れを防止

表形式にまとめることにより、処理の抜け漏れを低減できる

## 4.2.6 概念データモデル（ER図）（1）

### ■ 目的

- 実業務の管理対象や業務ルール等を分析・モデル化し、企業のビジネス活動の構造を理解する

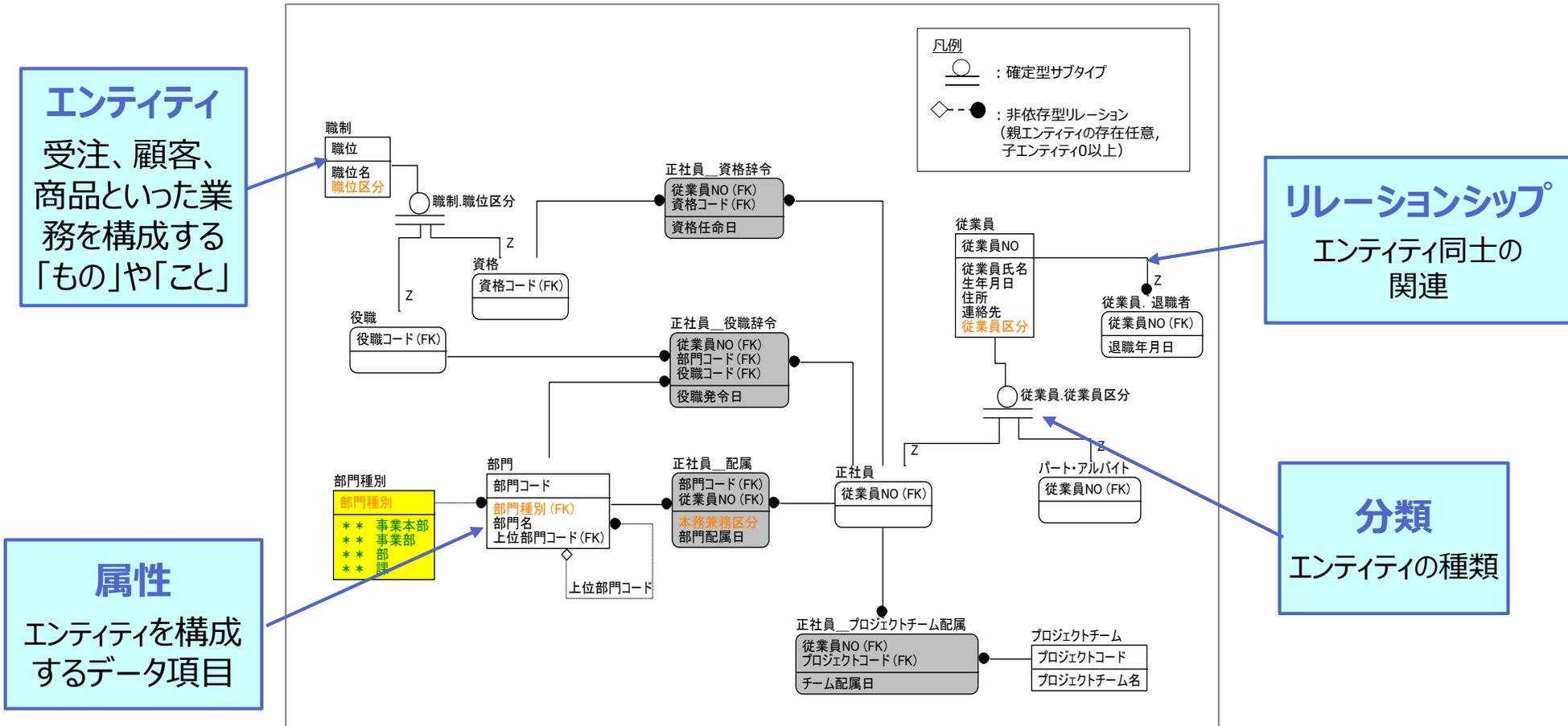
要件定義において概念レベルのER図で業務の実態を把握しておくことにより、業務の構造を正確に写像したデータモデルを描くことができ、基本設計ではそれをベースにテーブル設計ができるため、要件定義への手戻りを削減できる

### ■ 作成のポイント

- **As-Is概念データモデルをまず作成して、To-Beモデルとして変化させる**のが効率的である
- 現状を表すER図は**業務の実態、事実**に忠実に描く
- 具体的なデータを入れて検証してみる

# 4.2.6 概念データモデル (ER図) (2)

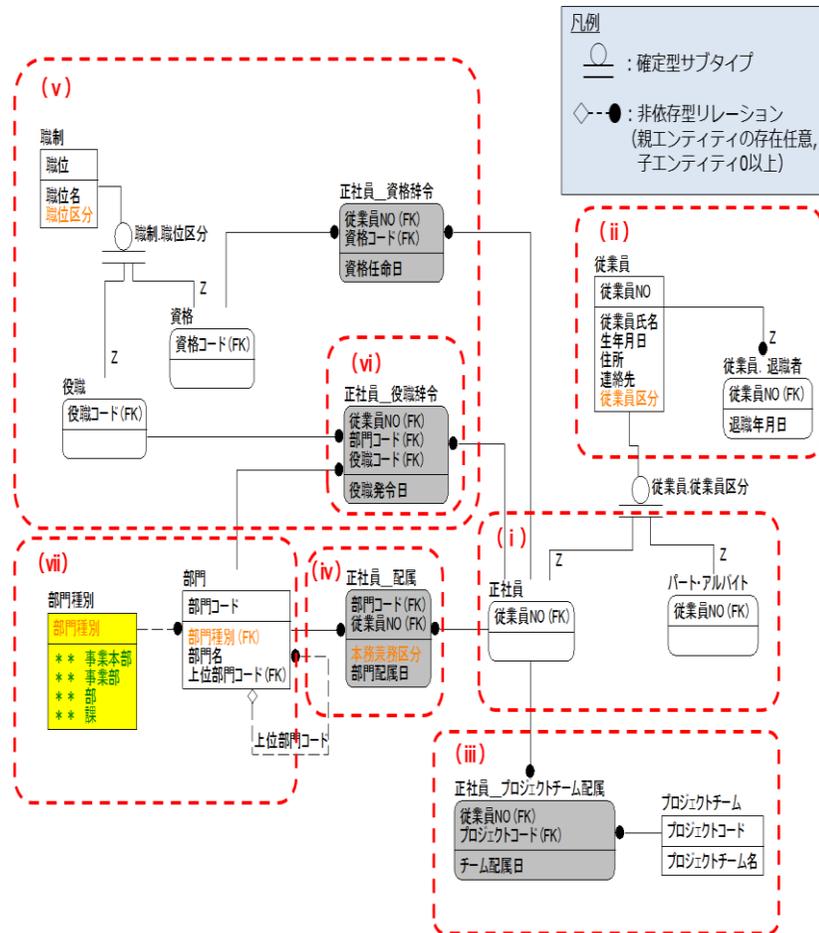
## ■ ER図の例 エンティティと関連と分類と属性



エンティティ、リレーションシップ、分類、属性を用いて業務のルールやビジネス活動を表現できる

# 4.2.6 概念データモデル (ER図) (3)

## ■ 概念データモデルには業務の実相を表わせる

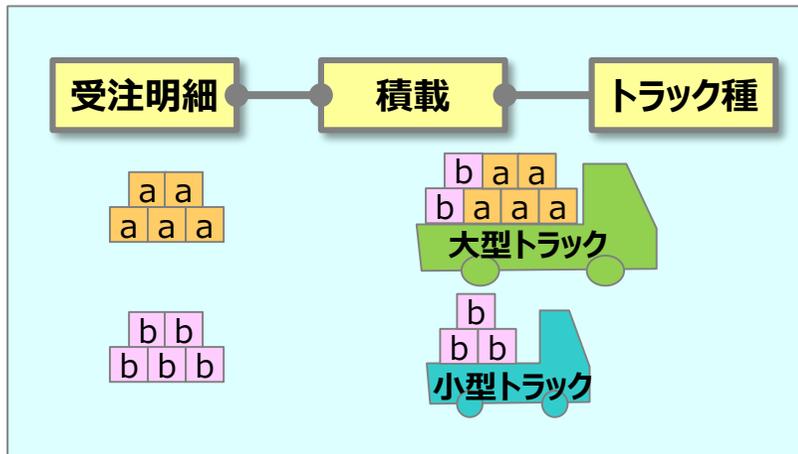


- i. 従業員には正社員とパート・アルバイトがある  
 正社員は職位の付与、部門への配属、プロジェクトへの配属があるが、パート・アルバイトはない
- ii. 従業員は現在在籍している従業員に加えて退職した従業員もスコープに含めて把握したい
- iii. 正社員は同時に複数のプロジェクトに従事できる  
 プロジェクトチームにはプロジェクトコードが付与され、プロジェクトコードでプロジェクトチームが管理されている
- iv. 従業員は本務に加えて他部門への兼務がある
- v. 職位には資格と役職があり、それぞれ辞令発令がある  
 役職とは、部長、課長、係長等を表し、資格とは技術スタッフ、1級、2級などを意味し、基本給のベースとなる
- vi. ある部門では部長だが、別の部門では課長というように部長兼課長のような兼任がある
- vii. 社内組織 (部門) は事業部の下に部、部の下に課等の階層関係が存在する

【コラム】～概念データモデル（ER図）を業務部門に確認する～

課題：概念データモデルを業務部門が作成するのは難しい

- 対策：■ **業務部門にヒアリング**しながら**システム部門が作成**する  
 ■ 概念データモデルそのものではなく**補助説明資料**を用意して確認しやすくする



絵を使用して説明



インスタンス（値）を使用して説明

## 4.2.7 エンティティ定義書／データ項目定義書（1）

### ■ エンティティ定義書の目的

- エンティティが共通認識できるように、エンティティ自体を説明する

### ■ エンティティ定義書作成のポイント

- エンティティの目的・用途を説明する
- エンティティにどのようなデータ項目が従属するかを定義する

エンティティ名	正社員__役職辞令
説明	正社員に発令した配属、役職任命の履歴を保持する

No.	属性名	主キー	データ型	桁数 文字数	説明
1	従業員NO	○	文字列	6	従業員を一意に識別する番号
2	部門コード	○	文字列	4	部門を一意に識別するコード
3	役職コード	○	文字列	2	役職を一意に識別するコード
4	役職発令日		日付	-	役職任命を発令した年月日

主要なデータ項目については業務部門が中心になってまとめる

項目の命名を整理して、同名異義、異名同義を排除する

## 4.2.7 エンティティ定義書／データ項目定義書（2）

### ■ データ項目定義書の目的

- 各データ項目の属性や管理ルールを定義する

### ■ データ項目定義書作成のポイント

- 次のような属性や管理ルールを定義する
  - データ属性（項目タイプ、桁数、小数点桁数、最大値、最小値など）
  - 初期値、Null値
  - 前提条件、制約条件 など

属性、ルールなど	説明
データ項目名	部門配属日
エンティティ	正社員_役職辞令
データ属性	日付型（他に、文字型、整数型、桁数、小数以下桁数、最大値、最小値など）
Null	不可
初期値	なし
前提条件 （制約条件など）	部門に配属されると、従業員NOと部門コードと同時に発生する
機密性	登録権限は人事にだけ付与
...	

Nullを許すかどうかと初期値の定義には特に注意する

## 4.2.8 CRUD図 (1)

### ■ 目的

- エンティティ、データ項目について、どの機能で作成、更新、消去されるかを確認することにより、その項目のライフサイクルを把握し、機能面、データ面での抜け漏れを検証する

### ■ 作成のポイント

- エンティティやデータ項目がどの機能で作成され（Create）、使用され（Read）、更新され（Update）、削除される（Delete）かを表す
- **エンティティレベルとデータ項目レベルを使い分ける**
  - エンティティレベルは大きな機能の抜け漏れや矛盾の発見に適する
  - データ項目レベルは項目が作成されていないのに更新されているなどの、データ管理上の矛盾を検証するのに適する

## 4.2.8 CRUD☒ (2)

### ■ CRUD☒ (エンティティレベル) の例

エンティティ	受注	生産手配	進捗管理	倉庫管理	出荷	売上計上	財務処理	削除処理
受注	C	U	U	U	U	U	U	D
出荷			U	U	C	C	U	D
⋮								

全体を  
鳥瞰

凡例 : C (登録、Create) 、 R (照会、Read) 、 U (更新、Update) 、 D (削除、Delete)

### ■ CRUD☒ (データ項目レベル) の例

データ項目	受注	生産手配	進捗管理	倉庫管理	出荷	売上計上	財務処理	削除処理
受注番号	C							D
受注数量	C							D
単価	C					U	U	D
受注金額	C				U	U	U	D
主要仕様 (色など)	C	U			U	U		D
生産数量		C	U	U	U	U		D
⋮								

部分を  
詳細

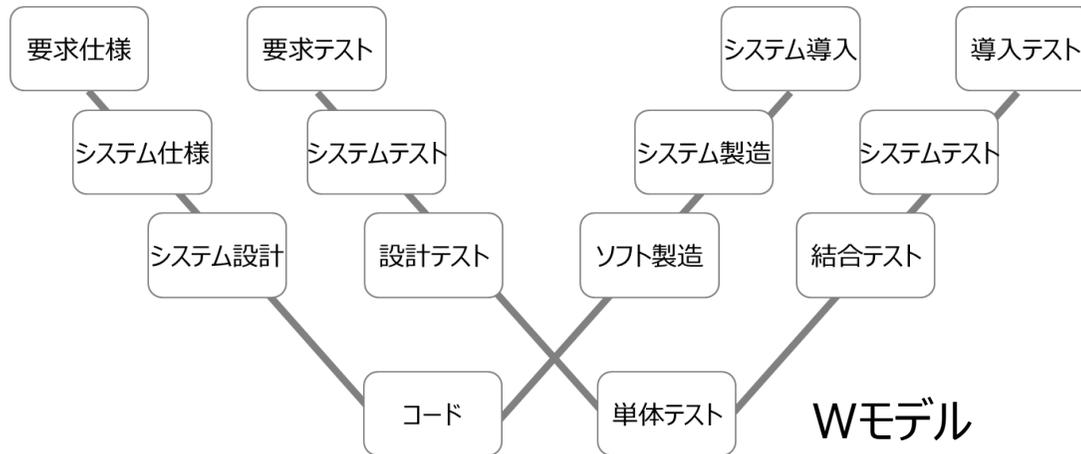
特定項目  
を追跡

凡例 : C (登録、Create) 、 R (照会、Read) 、 U (更新、Update) 、 D (削除、Delete)

## 4.2.9 総合テスト計画書 (1)

### ■ 総合テスト計画書の目的

- 要件定義で明確になった業務シナリオを、要件定義の検証プロセスである総合テストでもれなく実施するための準備をする



#### 要件定義の現状

- 要件定義ではテスト仕様書は作成されていない、作成に労力をかけられない
- 要件定義で明確にした業務パターンや業務シナリオがテスト仕様書に活かされていない

### ■ 作成のポイント

- 総合テスト仕様書の作成を要件定義行程から開始する
- ビジネスプロセス関連図や業務流れ図（業務フロー）を活用して総合テストシナリオを作成する

## 4.2.9 総合テスト計画書 (2)

### ■ 総合テスト計画書の構成例

総合テスト計画書(例)
1. 総合テストの目的
2. 総合テスト概要 (1) 対象、範囲 (2) 前提条件 (3) 環境 (ハード、ソフト、ネットワーク) (4) 期間、スケジュール (5) 体制
3. 総合テスト仕様書 (1) 総合テストシナリオによる一連のつながりの確認 (テスト) <ul style="list-style-type: none"> <li>・通常処理</li> <li>・例外処理</li> <li>・特殊処理</li> </ul> (2) サイクルテスト (日回しのテスト) (3) 締め処理 (日、月、年次) (4) 他システムとの連携テスト (5) 性能負荷テスト (6) 障害回復テスト (7) 移行リハーサルテスト (8) ユーザテスト
4. 総合テスト終了条件

テストシナリオは業務フローや業務シナリオマトリクスで明確になっている業務シナリオなので、それを利用する

## 4.2.10 システム移行計画書

### ■ 目的

- 現行システムから新システムへ、業務・データ・システムを問題なく移行する手順を計画する

### ■ 作成のポイント

- システムの全利用者を対象に全機能を一斉切り替えするか、機能別あるいは利用者別に順次本稼働するか、を計画する
- 新旧システムの並行稼働の有無、並行稼働の対象システムと期間、旧システムの停止時期と新システムへの移行時期を明確にしておく
- **要件定義の段階で**データ移行（コンバージョン）の対象・手段を検討する
- 障害発生時の対応や切り戻し判断を含めて手順を検討する
- 移行リハーサルの実施を計画に入れておく
- 要件定義段階で実施すべきことは、上記に関する要件を定義すること

## 4.2.11 運用・操作要件書

### ■ 目的

- 開発したシステムをシステム部門で実際に運用、操作でき、それによって業務を遂行できるための要件を定義する

### ■ 作成のポイント

- 運用部門を検討に巻き込む
- **要件定義の段階で**運用・操作要件を検討する
- 業務の運用要件も明確にする（障害時の代替手段など）
- システム運用部門の要件も各要件定義書に含めて記載してもよい
  - マスタメンテ画面・機能など運用上のシステム

## 4.2.12 非機能要件書（1）

### ■ 目的

- 業務機能を実現するために必要となるシステムの非機能要件を定義する

### ■ 作成のポイント

- **要件定義の段階で**以下のような観点で非機能要件を定義する
  - 処理データ発生量（業務量、トランザクション量、データ保存量）
  - レスポンスタイム（通常時・ピーク時・将来予想、前提とする機器構成）
  - 使用性（使い勝手）
  - 稼働率、稼働品質（システムの特質に応じた稼働率、稼働品質の評価）
  - セキュリティ（データや業務へのアクセス権限管理、業務の監査ログなど）
  - 保守性（機能追加を容易にするシステム開発手法の選択）
  - 移植性（広く普及している標準的な技術の組み合わせで構築）
- トレードオフを明らかにし、関係部門も加えて妥当性を評価する

## 4.3.1 要件定義成果物のレビュー（1）

### ■ 目的

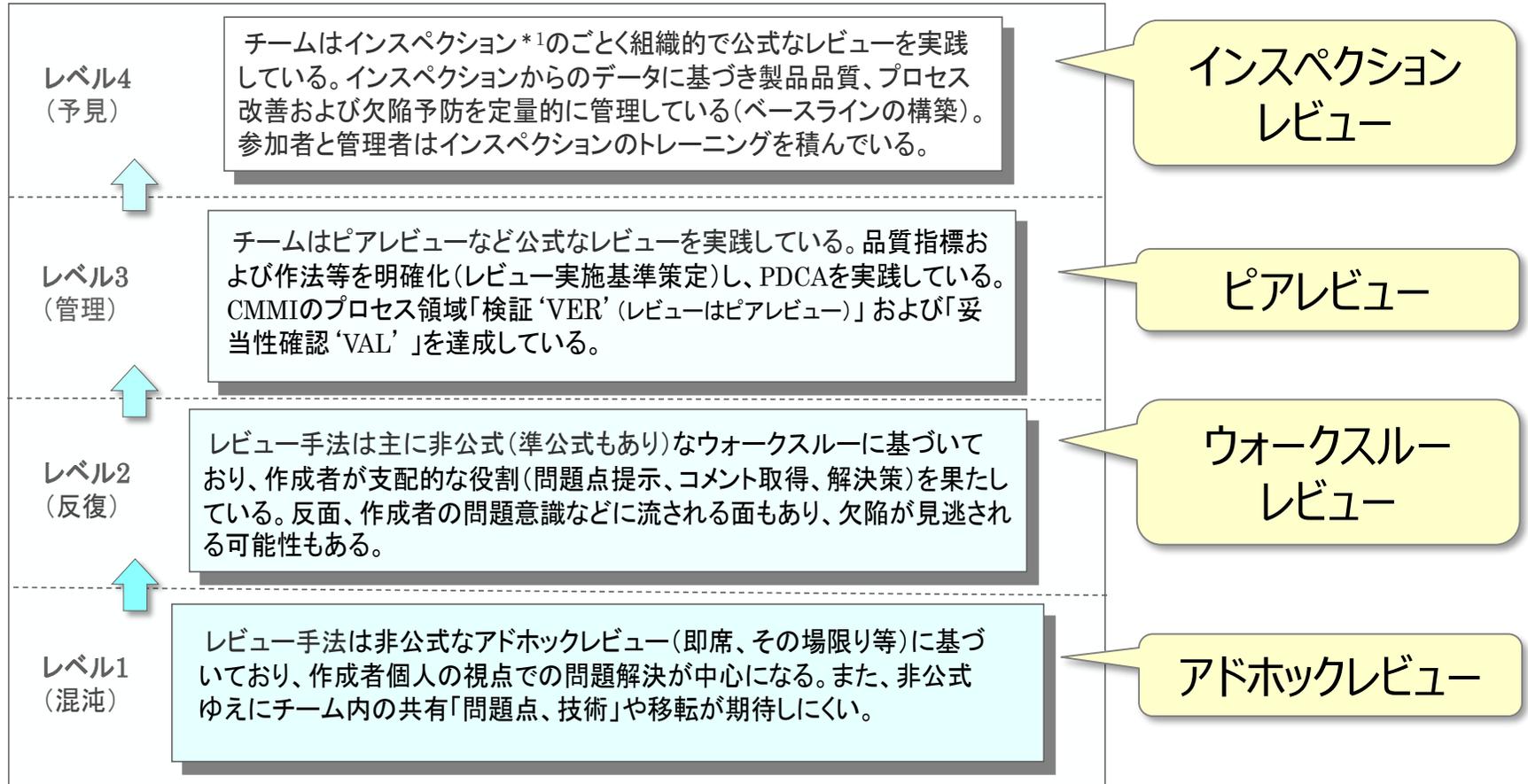
- 要件定義成果物の抜け・漏れや、成果物間の整合性不備などを、**第三者による熟視**を主体とした**公式なレビュー**による検査によって摘出する（インスペクションレビューの勧め）

### ■ 留意点

- 高信頼ソフトウェアの要求および要件定義成果物のレビュー方法を検討する場合、インスペクションは重要な手法になる
- インスペクションの目的
  - 成果物（ソフトウェア要素）が基本仕様、指定された品質属性、顧客要求を達成しているかの検証
  - 成果物（ソフトウェア要素）が関連する基準、規制、規則、計画、手順に適合しているかの検証
  - 発見された欠陥とインスペクションに使用した時間の評価指標の供給
  - 表現方法の良し悪しや文体に対する検査は、それ以前に実施しておく

# 4.3.1 要件定義成果物のレビュー（2）

## ■ レビューの種類（5.7参照）



\* 1 : ANSI/IEEE 標準 1028-1998で体系化され定義されており、厳格で公式なレビュー方法です。他の手法に比べ、欠陥摘発にもっとも効果が高い手法の一つです。

# 4.3.1 要件定義成果物のレビュー (3)

## ■ インспекションでの各自の役割

役割	作業内容など
進行・まとめ役 (Moderator)	<p>インспекションの主催者である。</p> <p>インспекションを熟知している必要がある。インспекション・メンバーの選抜やスケジュール調整、成果物の受領と配布、さらには作業進捗の管理を行う。</p> <p>計画段階で、ある条件が揃った場合にインспекションの中断か継続可能かのチェックポイントを設けておく。</p> <p>最終的には当該インспекションの結果報告責任者になる。</p>
レビューを行う役 (Inspector)	<p>欠陥検出の担当で、当該チームの作業内容について既知であるか否かを問わず、高い欠陥発見能力を持つ人が望まれる。</p> <p>インспекションでは、対象成果物ごとの関連性を考慮して、レビューを行う役などを決める。</p> <p>インспекションはオーバーヘッドなど非効率性を鑑み、最適人数を決定する。(経験則として3名~4名まで)</p> <p>インспекションに関する正式な教育を受けていることが必要である。</p>
説明役 (Reader)	<p>進行に合わせて対象成果物の該当部分を他に説明する。一般的には作成者が担当する。</p>
記録役 (Recorder)	<p>指摘された欠陥と問題を分類、記録する。</p>
作成者 (Author)	<p>インспекション対象に関わる成果物の作成者または保守者が該当する。</p>

第三者の目による厳格なレビュー

誤字脱字や形式の不整合など、一見して分かる欠陥は、レビュー前になくしておく

インспекションにおいて**作成者**が担う役割はここだけ

## 4.3.2 要件定義における未決定事項の取扱い

### ■ 目的

- 未決事項の重要度を評価し、残課題として次工程に進んでも問題ないかどうかを判断する

### ■ 留意点

- 未決事項を整理・評価し、その軽重を判断して、条件付きで次工程に入るかどうかの判断と、未決を残して次工程に進んだ場合の**リスクを関係者間で共有**できるようにする
- **要件が定義できない事案は廃棄する**、と最初に宣言し、未決を残さない、どうしても未決を残す場合には、次工程の初期に完了させる目処があることを条件にする

- 要件定義成果物を高い品質で作成するために注意すべきこと
  - 最初に成果物（作成するドキュメント）と記載レベルを決定する
  - 個々のドキュメントの品質向上（抜け漏れ曖昧の低減）に注力する
  - レビュー能力を組織レベルで向上させる

**要件定義成果物の品質は、各ドキュメントの書式だけでなく、その目的（何に使用するのか、そのためには何をどう記載するのか）をしっかりと理解して作業することによって保たれる**