わかるシステム可視化技法

土俵際で相撲をとるのはやめよう

Copyright 2018 (株)バリューソース

- 株式会社　バリューソース
  - 神崎　善司

- 普段は
  - システム企画・要件定義などの支援
  - 既存システムの可視化支援
  - セミナー開催（要件定義、モデリング）
  - ビジネス系のシステムが対象

- 要件定義手法の開発

  –RDRA

  - **Relationship driven requirement analysis**

わたしは・・・

このようなことはありませんか1 （ユーザ企業）

- システムの実装は分からない
  - 実装より仕様をはっきりさせましょう

RDRAでシステムの仕様を表現しましょう

- 今のシステムを表す資料がない
  - 案件の資料ばかり
  - システム仕様を作っても維持できない

システム仕様を維持する仕組みを作る

- システムの規模が大きく知識が属人化している
  - キーマンはいつも忙しい
  - ドキュメントではなくコミュニケーションのベースを作る

類推できるレベルで関係者で共有しましょう

- 保守開発の期間とコストが上昇している
  - 影響度分析に時間とコストがかかり過ぎている

影響度を分析できる仕組みをもつ

Copyright 2018 (株)バリューソース

このようなことはありませんか２（開発案件１次受け）

- 開発ボリュームは見えたかね？
  - まだ、プロジェクト開始２週間です

RDRAには短期間にボリュームを可視化する仕組みがあります

- このシステムは複雑かね？
  - 細かい仕様まではまだ？

RDRAはビジネスルールが表現でき、複雑度はビジネスルールで測る

- スケジュールはいつできる？
  - 資料がありすぎて（無さ過ぎて）スコープが見えてません

ボリュームと複雑度が分かればスケジュールは立てられる

- 開発期間の短縮・品質の向上・コスト低減を忘れずに！
  - 耳にタコです　どうすればいいんですか

まずは可視化し重要なものを明確にしてから作戦を立てる

- 要件定義工程が混乱して期日までに終わりません
  - みんながいろんなことを言ってまとまりません

要件定義には決め方があります

Copyright 2018（株）バリューソース

このようなことはありませんか３（開発案件２次受け）

- 仕様書が沢山来たぞ
  - フォーマットは一致しているが内容がバラバラだ！
  - 粒度もバラバラだ！
  - そもそも仕様の精度が悪かったら手戻り必至だぞ
  - 後で仕様の不備が原因で仕様変更があっても、仕様の追加変更にならない
- どうやって短期間に仕様の精度を確認するか
- 設計・実装過程での影響度分析がうまくできない

RDRAの枠組みにはめれば精度を測れます

分析可能な可視化を行う

Copyright 2018（株）バリューソース

まとめると

- 短期間でボリュームを捉えられるようにする
- 合意を得る仕組みを持ち関係者で共有する
- 類推できる粒度でメンテナンス可能なボリュームに保つ
- ビジネスルールで複雑度を測れるようにする
- 保守開発の影響度を測れるようにする
- 精度を向上させる仕組みをもつ

Copyright 2018 (株)バリューソース

どうやって表現するのか
RDRAとは　何か
大事なことは何か

RDRA2.0 Relationship Driven Requirement Analysis

Copyright 2018 (株)バリューソース

サンプルを使ってシステムの読み方を説明

システムの読み方　SoraDo

Copyright 2018 (株)バリューソース

システムを表現するための考え方

Copyright 2018 (株)バリューソース

- 短期間でボリュームを捉えられるようにする
- 合意を得る仕組みを持ち関係者で共有する
- 類推できる粒度でメンテナンス可能なボリュームに保つ
- ビジネスルールで複雑度を測れるようにする
- 保守開発の影響度を測れるようにする
- 精度を向上させる仕組みをもつ

RDRAで問題を解決する

Copyright 2018 (株)バリューソース

まとめ

Copyright 2018 (株)バリューソース

Copyright 2018 (株)バリューソース

どうやって表現するのか

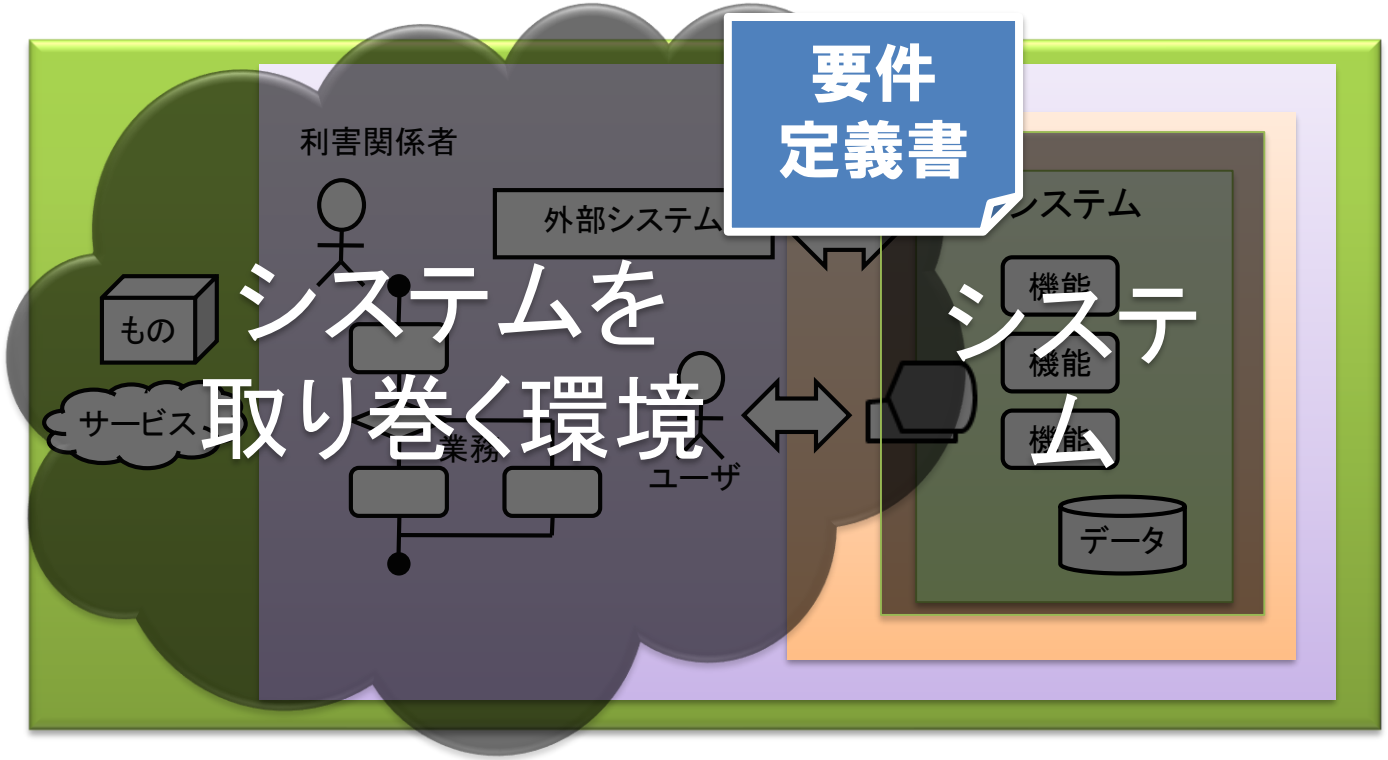
RDRAとは　何か

大事なことは何か

RDRA2.0 Relationship Driven Requirement Analysis

Copyright 2018（株）バリューソース

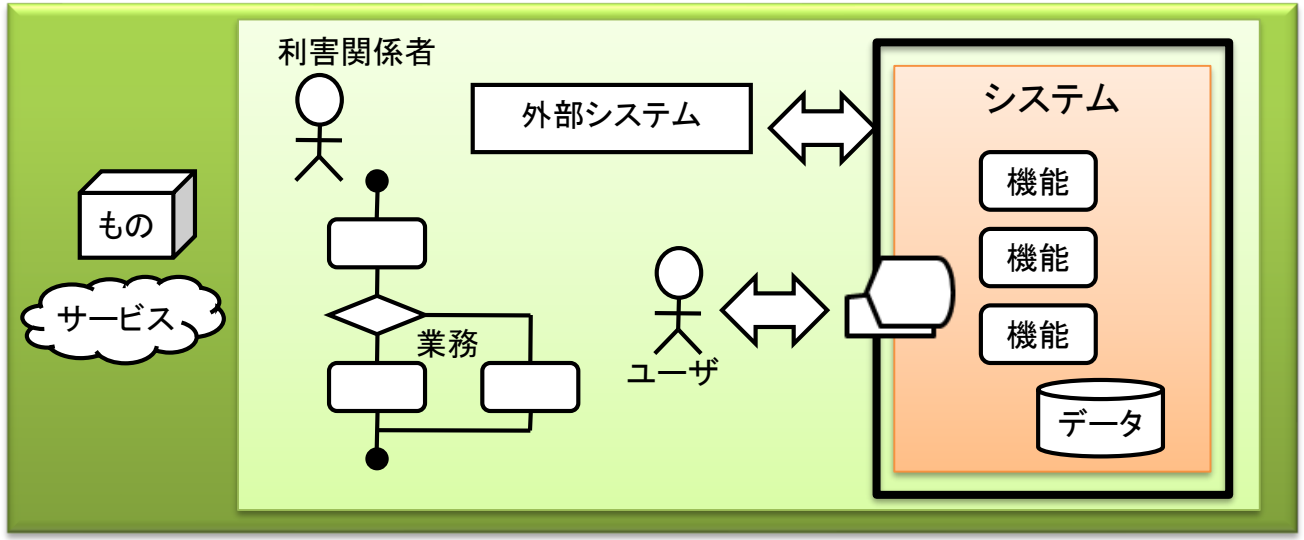
要件定義には何を定義すればいいのか



RDRAでは「要件定義の対象をシステムとシステムを取り巻く環境」と考える

Copyright 2018 (株)バリューソース

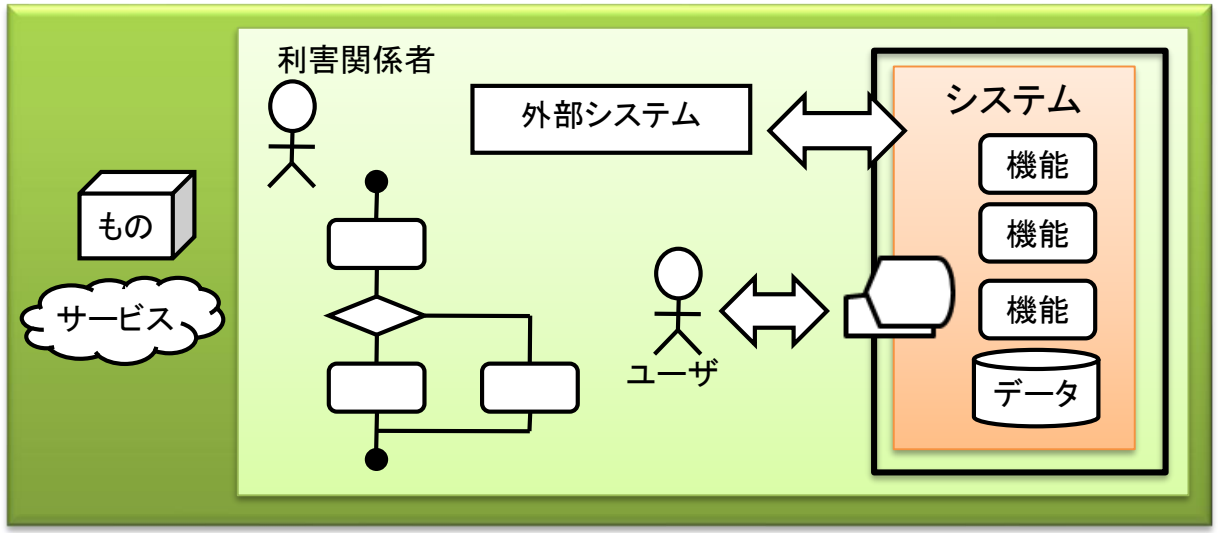
要件定義では何が定義されないといけないのか



- このシステムの目的（価値）は？
- どのような人、外部システムと関わるのか？
- どのようにシステムは使われるのか？
- システムとの接点は？
- その時の入出力情報は？
- システムに必要な機能は？
- その機能が使用するデータは？

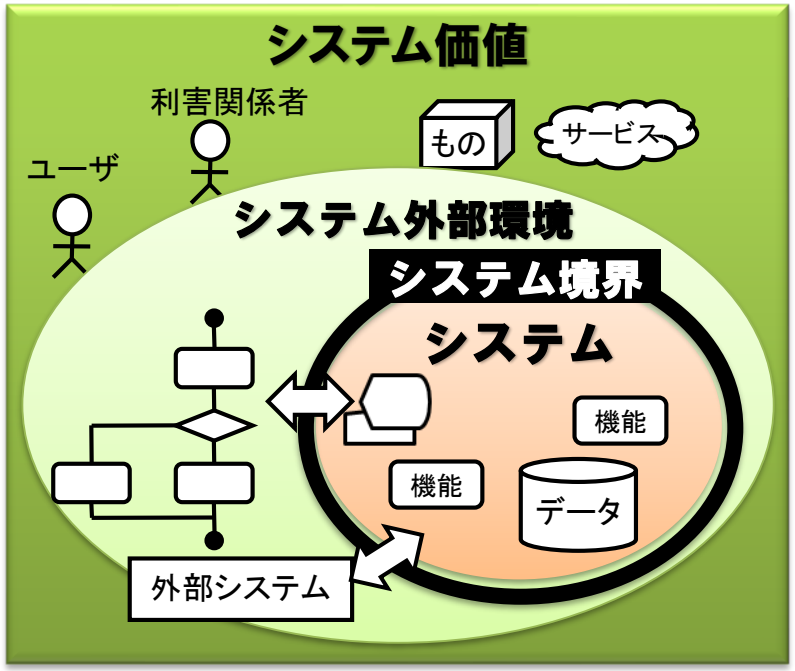
Copyright 2018 (株)バリューソース

要件定義の構造を定義する



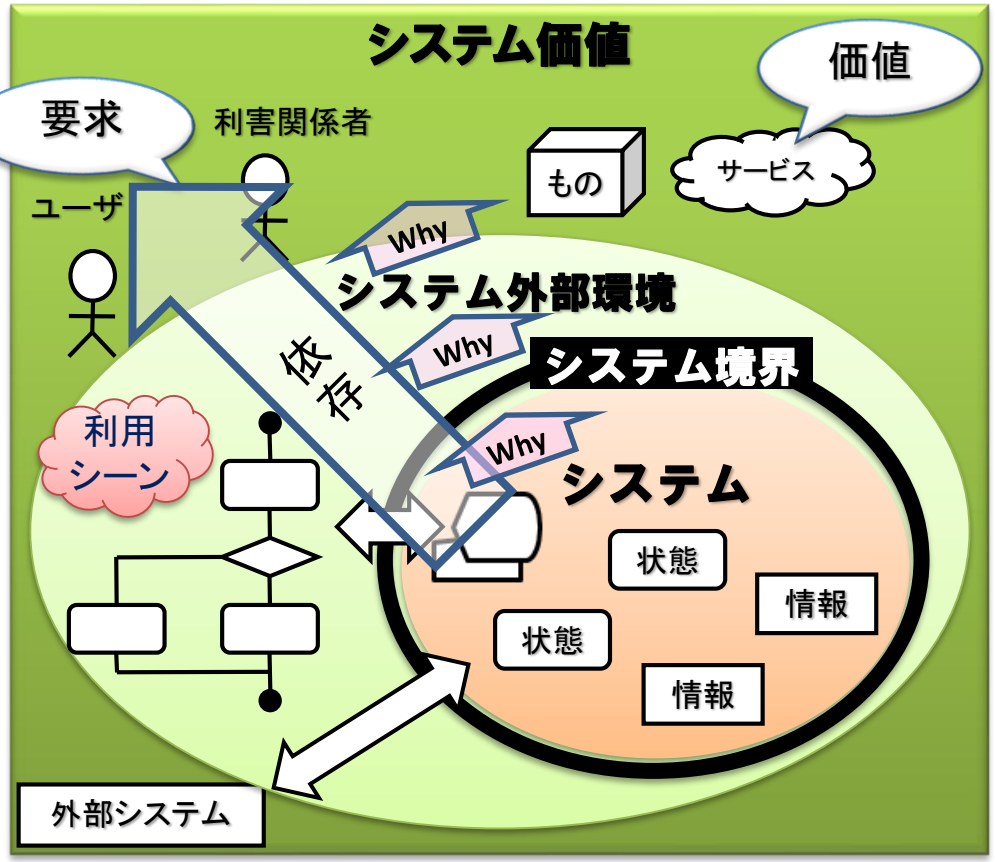
要件定義の構造

- システム価値
- システム外部環境
- システム境界
- システム



Copyright 2018 (株)バリューソース

要件分析の枠組み　構成要素



1.【システムの価値】を捉える

- 対象業務に関わる人を洗い出す
- 関係する外部システムを洗い出す
- 要求を捉える

2.【システム外部環境】を捉える

- 対象業務の流れを捉える
- 対象業務に関わる概念を明らかにする
- システムの利用シーンを明らかにする

3.【システム境界】を捉える

- ユーザインターフェースを明らかにする
- 外部システムとのインターフェースを明らかにする

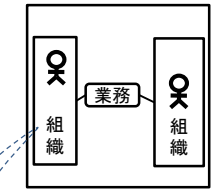
4.【システム】を定義する

- システムが扱う情報（概念、ことば）構造を明らかにする
- 情報が取りうる状態を明らかにする

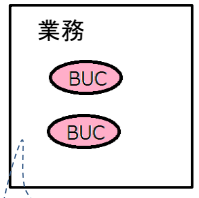
Copyright 2018（株）バリューソース

RDRA2.0　何をどう表現するのか

ビジネス可視化

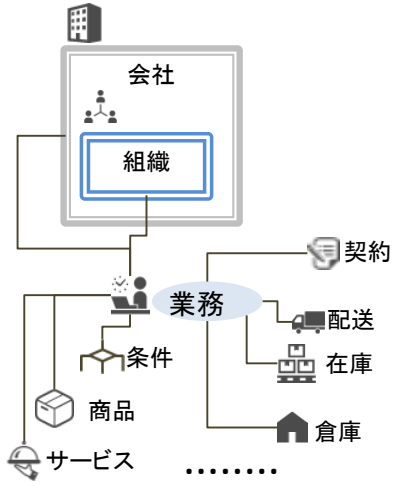


ビジネス
コンテキスト



ビジネス
ユースケース

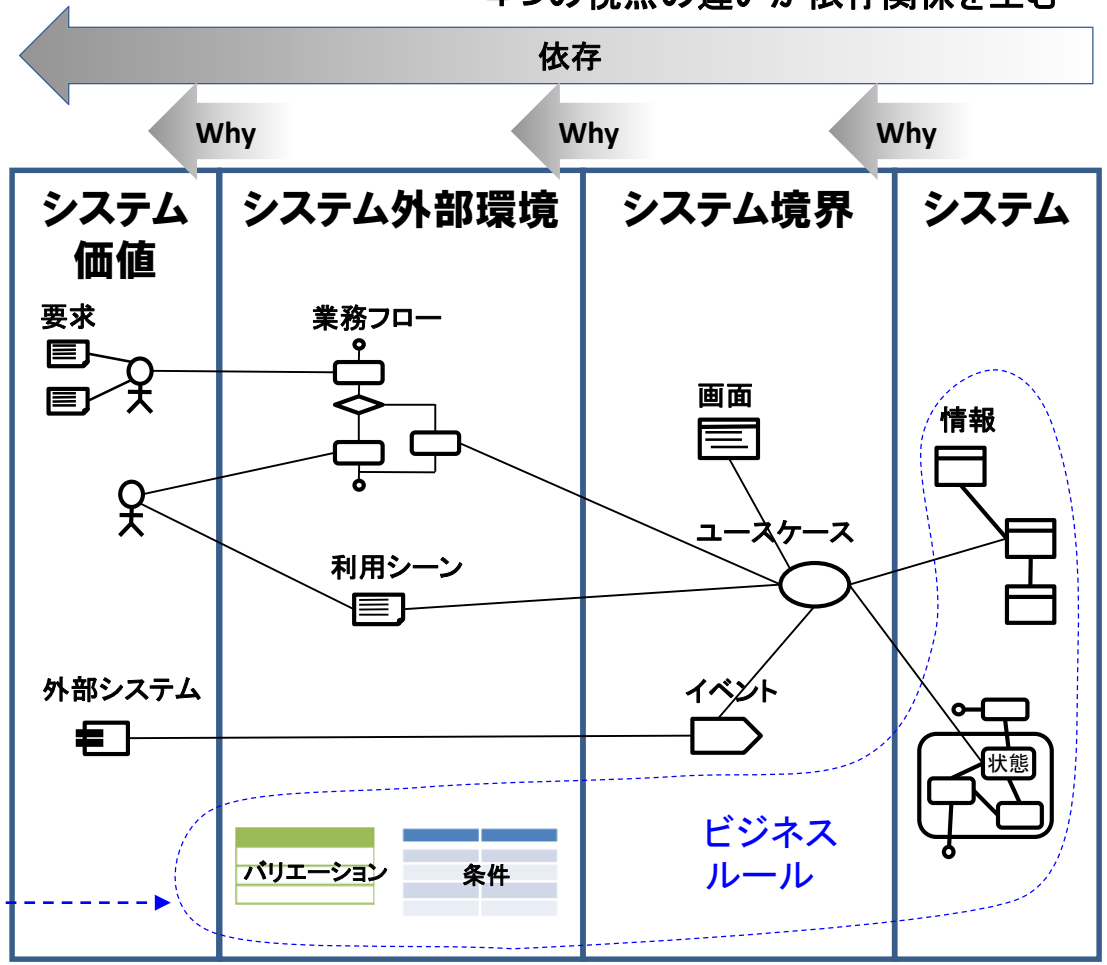
リッチピクチャー風アイコン



ビジネスのイメージが湧くよう
にわかりやすいアイコンを使う

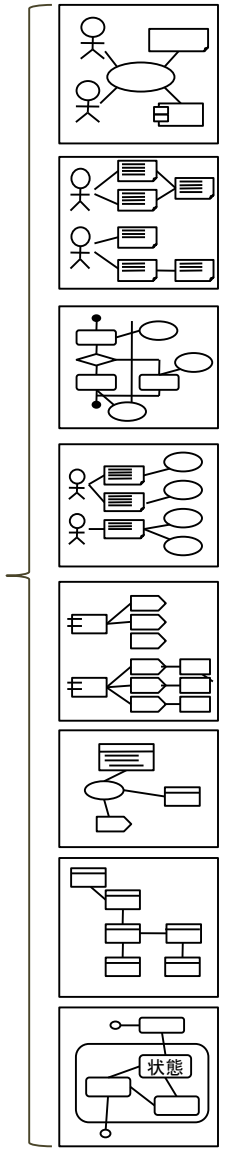
RDRAの基本構造

4つの視点の違いが依存関係を生む



・決められたアイコンで表現する
・ビジネスコンテキストの登場人物からバリエーションを洗い出す

ダイアグラム



システム
コンテキスト

要求モデル

業務フロー

利用シーン

イベントモデル

UC複合モデル

情報モデル

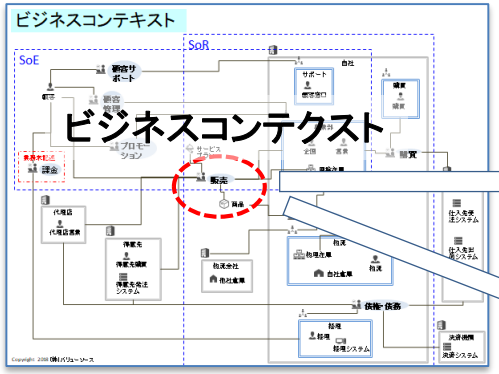
状態モデル

ボリュームに合わせてダイ
アグラムの書き方を変える

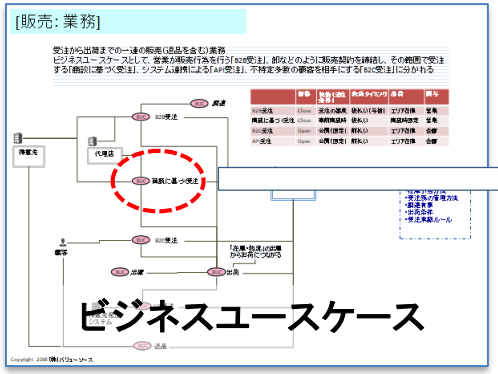
Copyright 2018（株）バリューソース

システムのダイアグラム構造

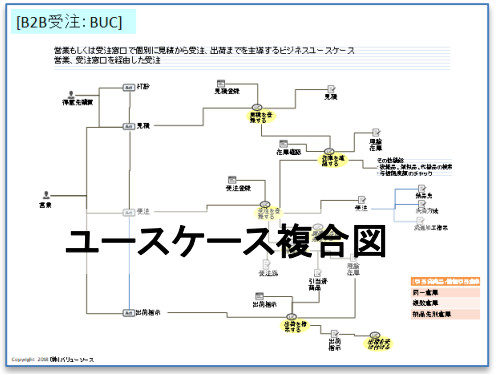
システムの全体像



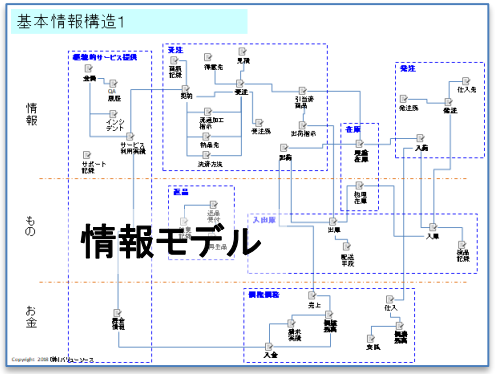
業務の構成をビジネスユースケースとして示す



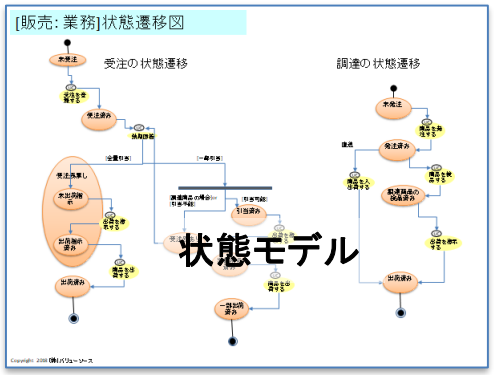
ビジネスユースケースの構成をユースケースを中心に示す



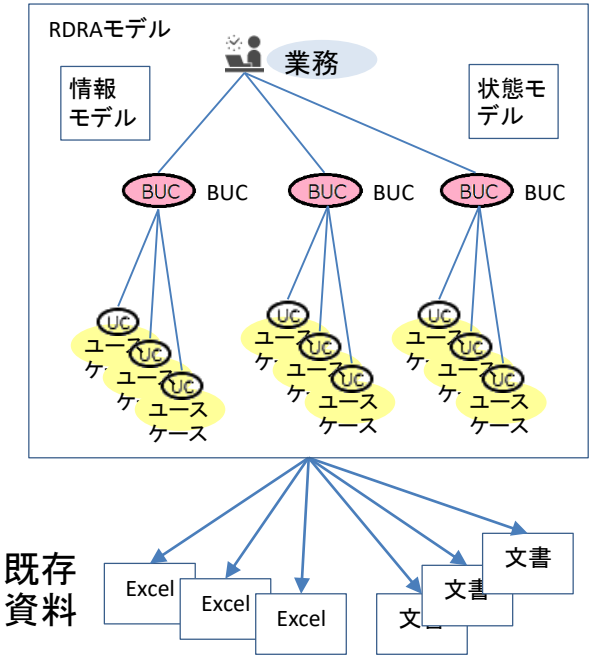
全体の情報構造を示す



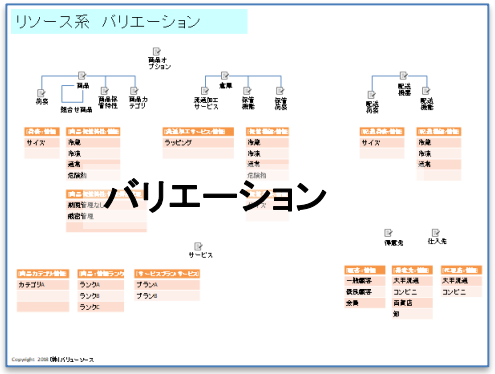
状態を表す言葉の関係性を表す



階層化された文脈で体系化する



バリエーションを示す



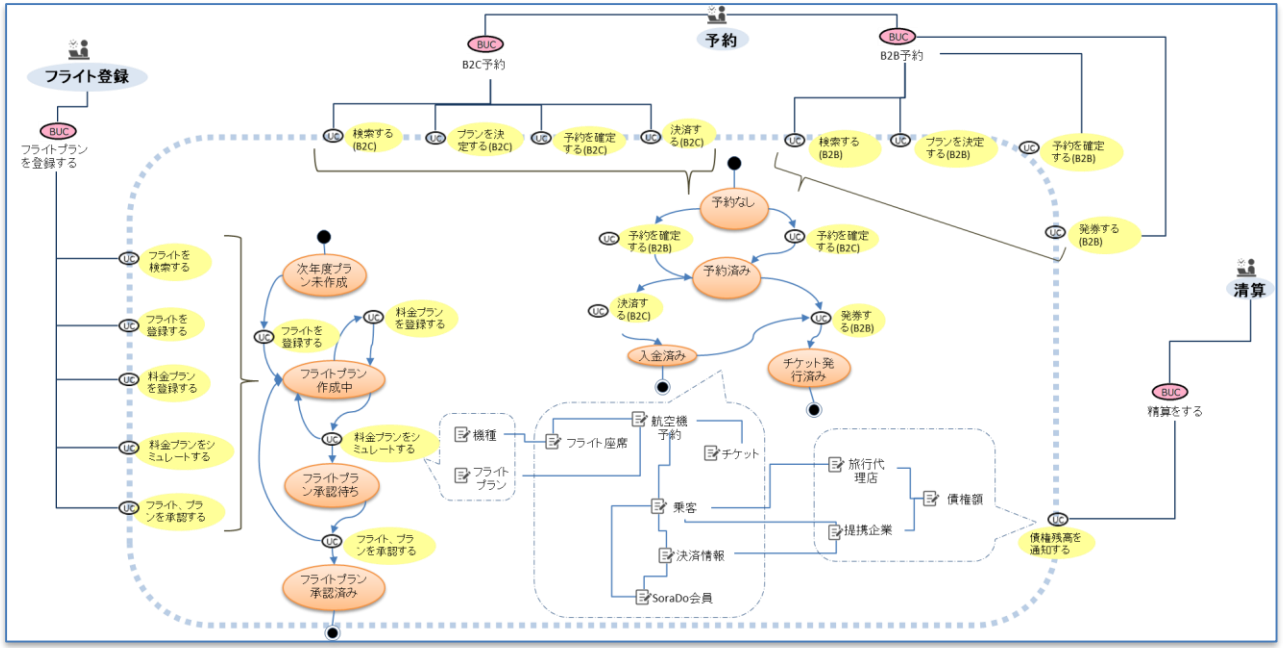
Copyright 2018 (株)バリューソース

RDRAとよくある方式との違い

- よくある方式:
  - 結果を一覧表の作成を目的化する
  - 行毎に詳細な資料を作成する
  - 情報のつながりは各担当者の頭の中

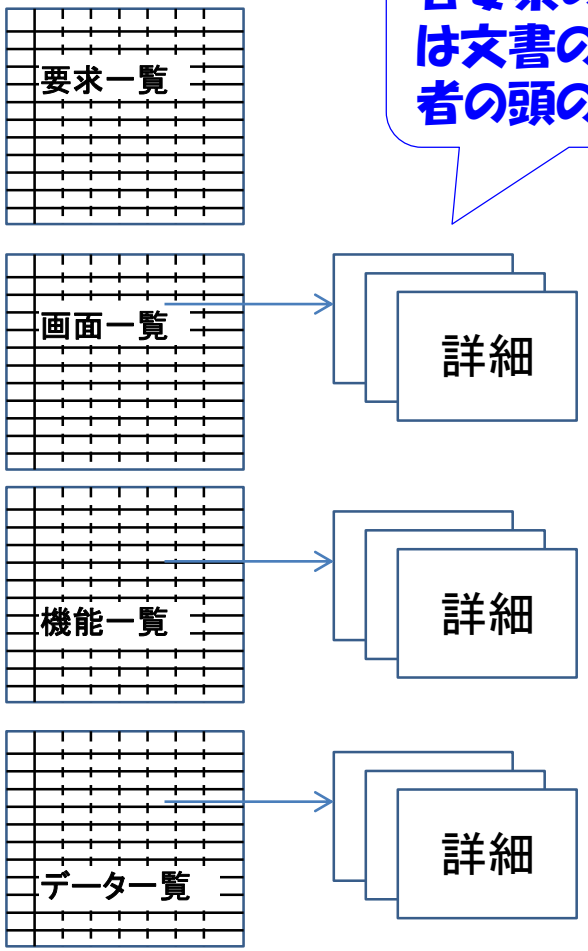
- RDRA
  - 情報のつながりを可視化
  - 結果として一覧が出力される

RDRA



出力

よくある方式



各要素のつながりは文書の中か担当者の頭の中

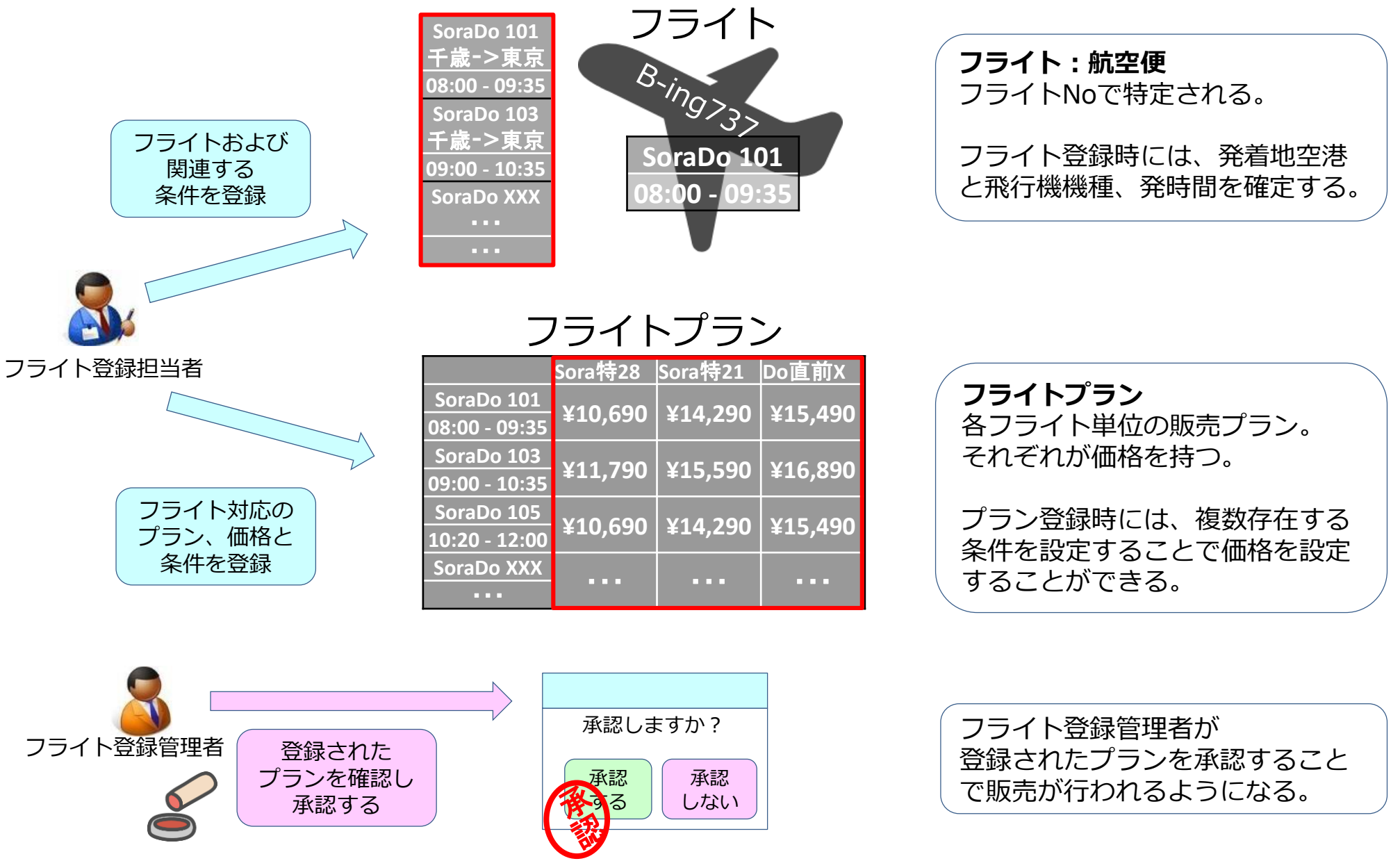
Copyright 2018 (株)バリューソース

サンプルを使ってシステムの読み方を説明

システムの読み方　SoraDo

Copyright 2018 (株)バリューソース

「フライト登録」に関わる業務イメージ



フライト：航空便
フライトNoで特定される。

フライト登録時には、発着地空港と飛行機機種、発時間を確定する。

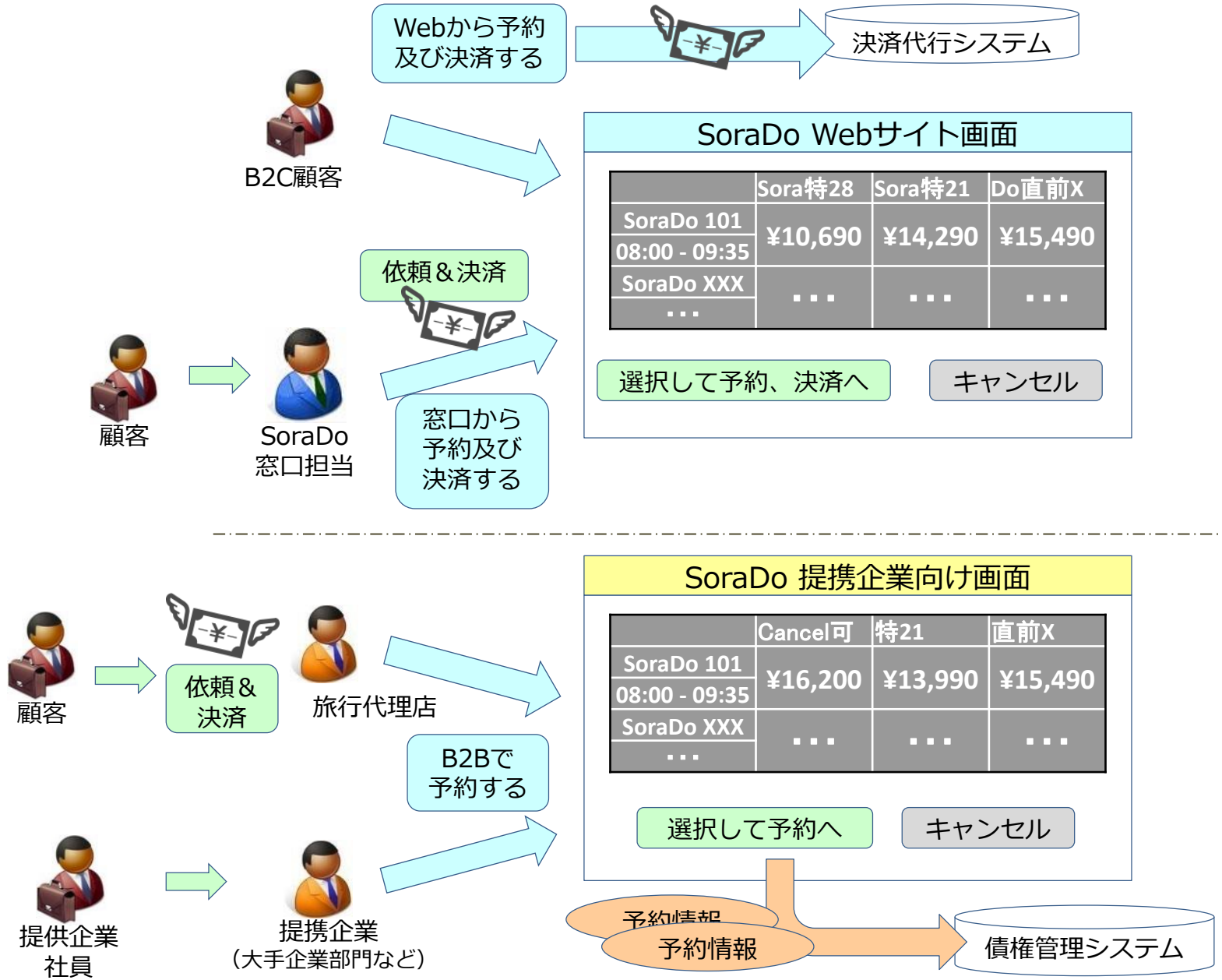
フライトプラン
各フライト単位の販売プラン。
それぞれが価格を持つ。

プラン登録時には、複数存在する条件を設定することで価格を設定することができる。

フライト登録管理者が
登録されたプランを承認することで販売が行われるようになる。

Copyright 2018（株）バリューソース

「予約」に関わる業務イメージ



Webからの予約
SoraDoWebサイト画面でB2C
顧客がフライトを予約、決済する。

顧客の決済用にクレジットカード対応の
決済代行システムを使用。

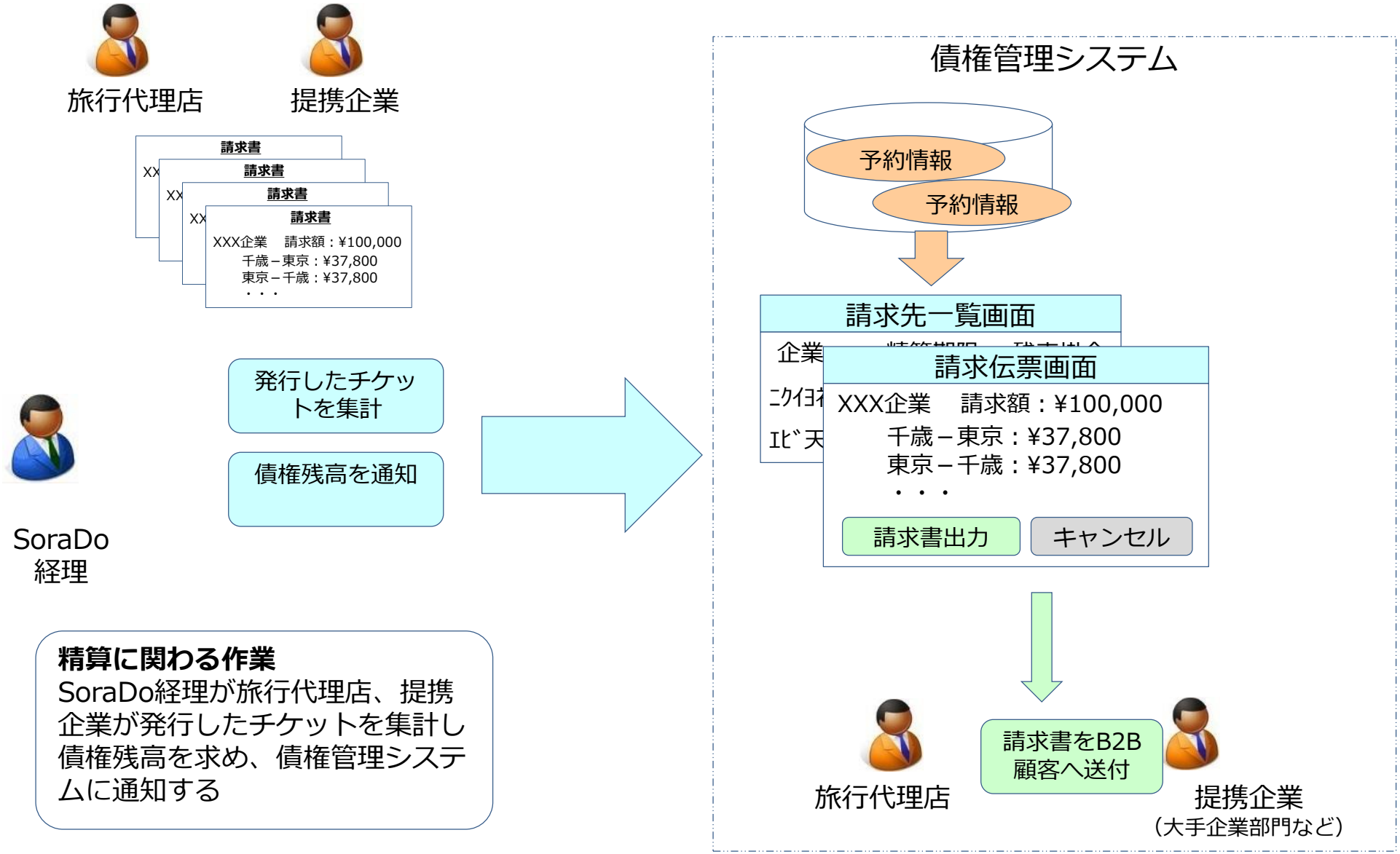
窓口からの予約
B2C顧客がSoraDo窓口に直接訪問して
予約を実施する。
販売価格はWeb予約と同じ。
窓口にて直接決済対応が行われる。

B2C顧客が旅行代理店経由で予約
大手企業の出張対応部門の予約

販売価格が多少割引となる。
※大口顧客となるため
また、予約情報は債権管理システムに記
録して、月単位で精算する。

Copyright 2018（株）バリューソース

「精算」に関わる業務イメージ

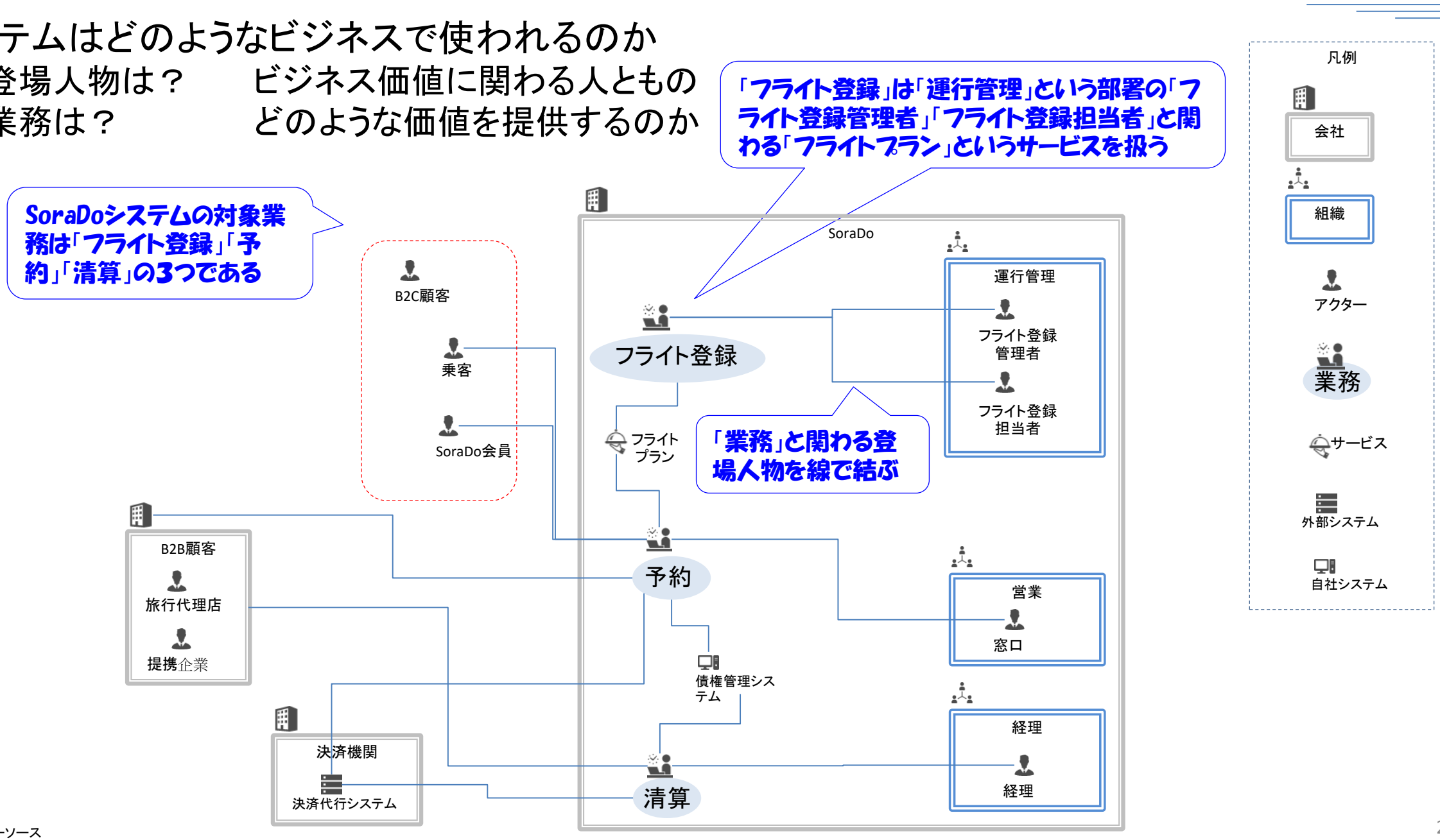


精算に関わる作業
SoraDo経理が旅行代理店、提携企業が発行したチケットを集計し債権残高を求め、債権管理システムに通知する

Copyright 2018（株）バリューソース

ビジネスコンテキスト

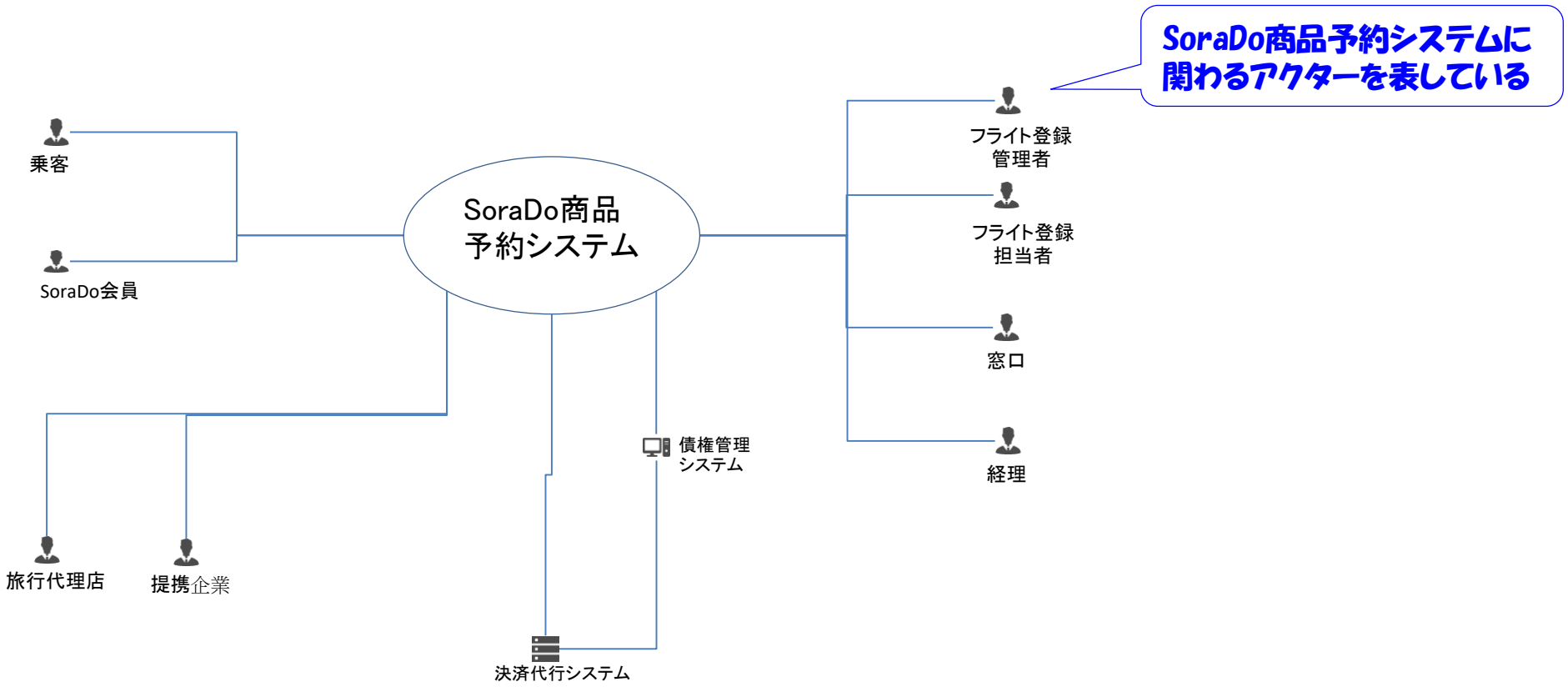
- システムはどのようなビジネスで使われるのか
  - 登場人物は？　　ビジネス価値に関わる人ともの
  - 業務は？　　どのような価値を提供するのか



Copyright 2018 (株)バリューソース

システムコンテキスト

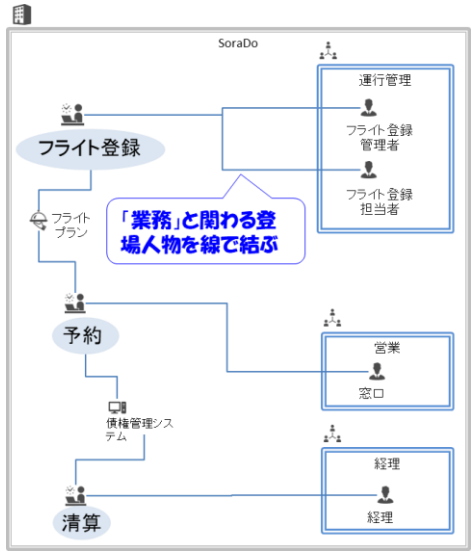
- システムコンテキストが表すもの？
  - システムに関わるアクターと外部システム
    - システムに価値をもたらすものを表す



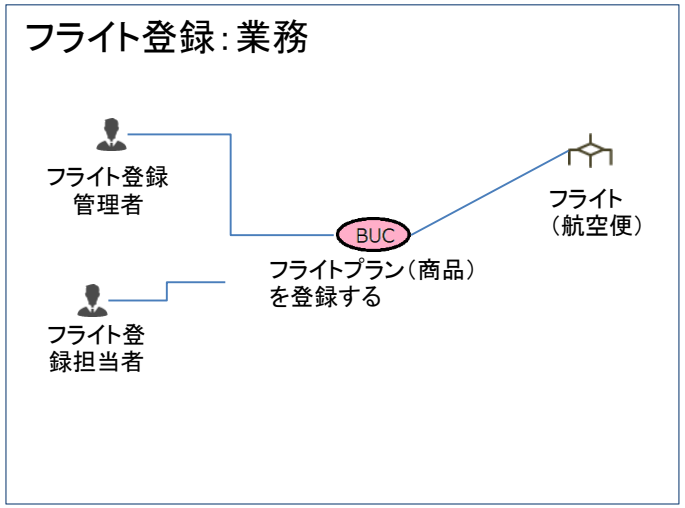
Copyright 2018 (株)バリューソース

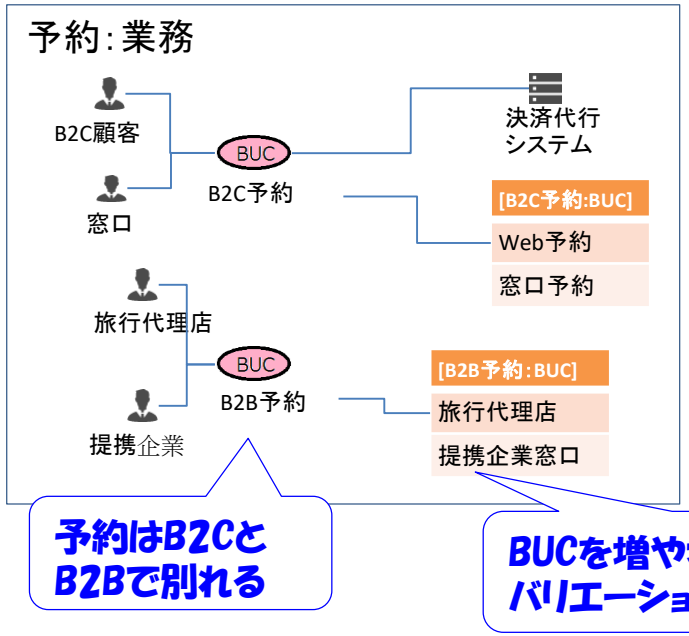
ビジネスユースケース

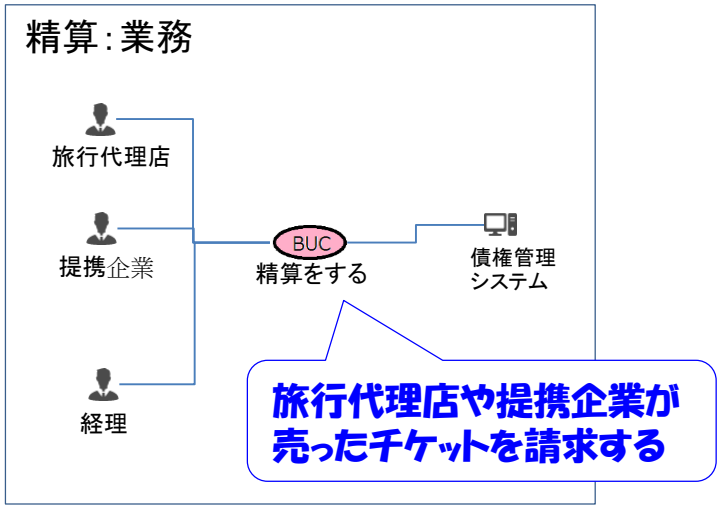
ビジネス
コンテキスト

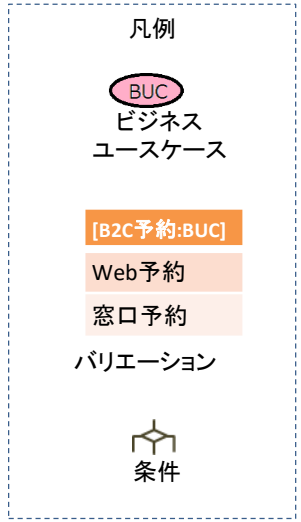


- ビジネスユースケースは何を表すのか
  - 「業務」をビジネスユースケースで表す
    - ビジネスコンテキスト内の「業務アイコン」
  - 同じ業務でも関わる登場人物が異なると仕事が変わる。その塊をビジネスユースケースで表す
  - BUCを増やすとシステム規模が大きくなる傾向がある
    - バリエーションを使って記述量を減らす





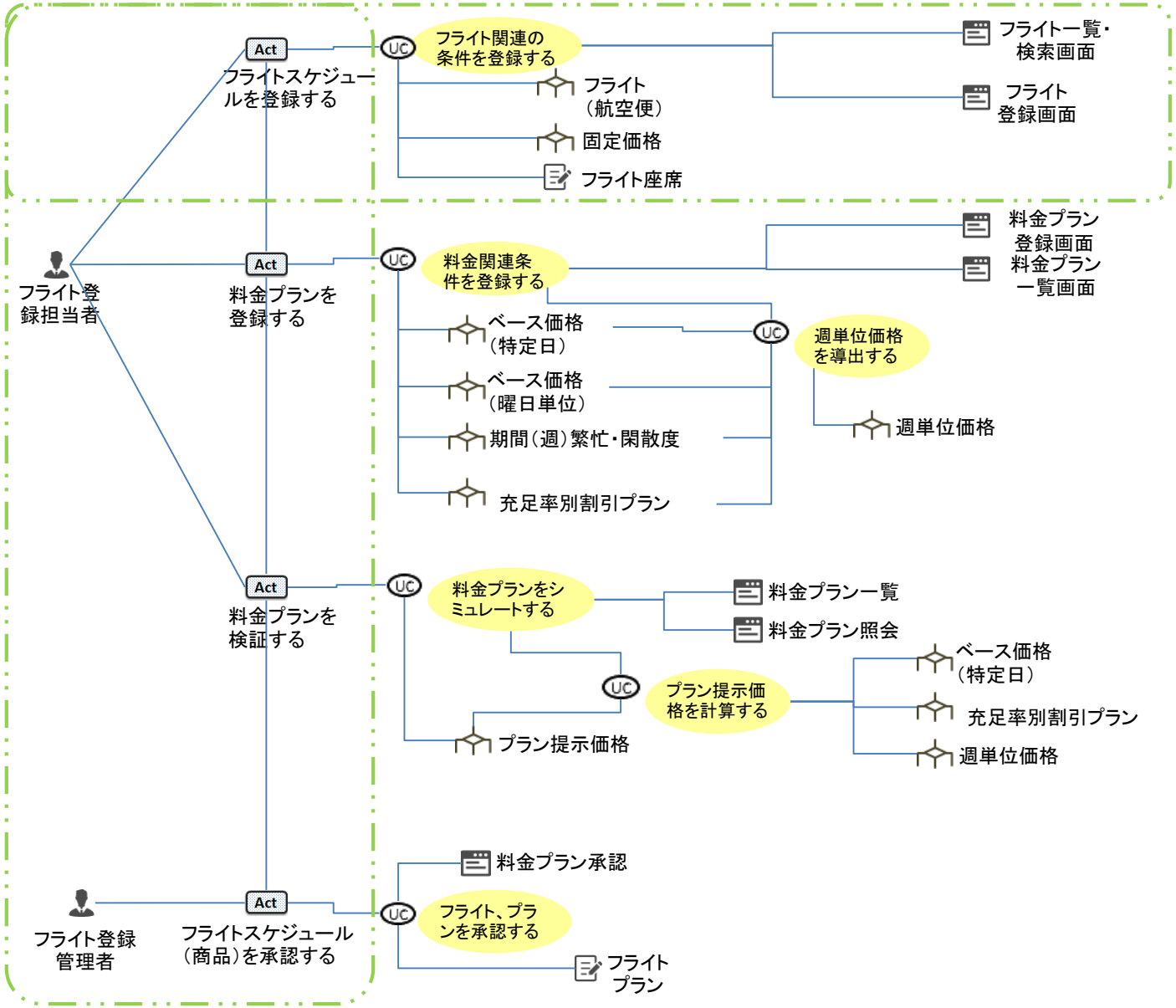




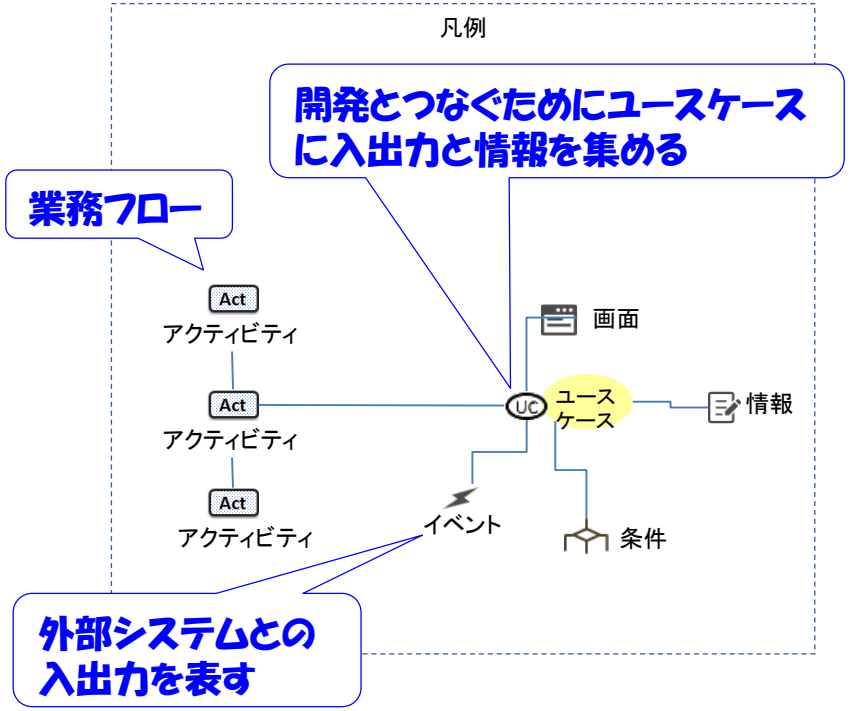
Copyright 2018 (株)バリューソース

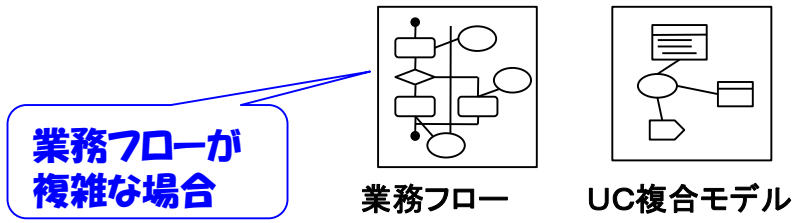
ユースケース複合図

フライトプラン(商品)を登録する:BUC



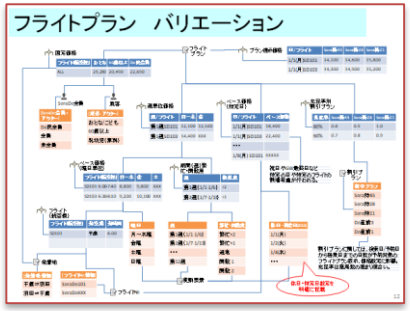
- ユースケース複合図は何を表すのか？
  - ビジネスユースケースを業務フローを起点にユースケース単位で記述

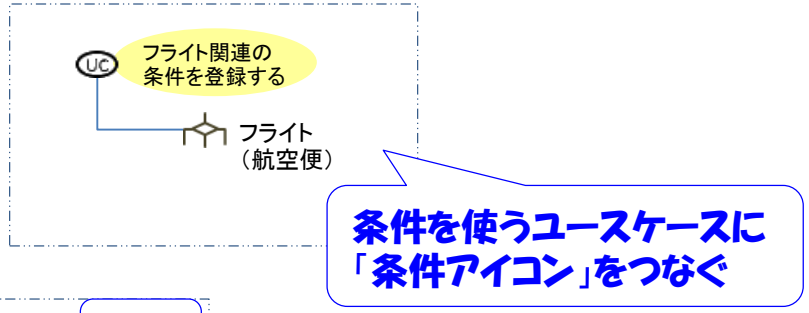


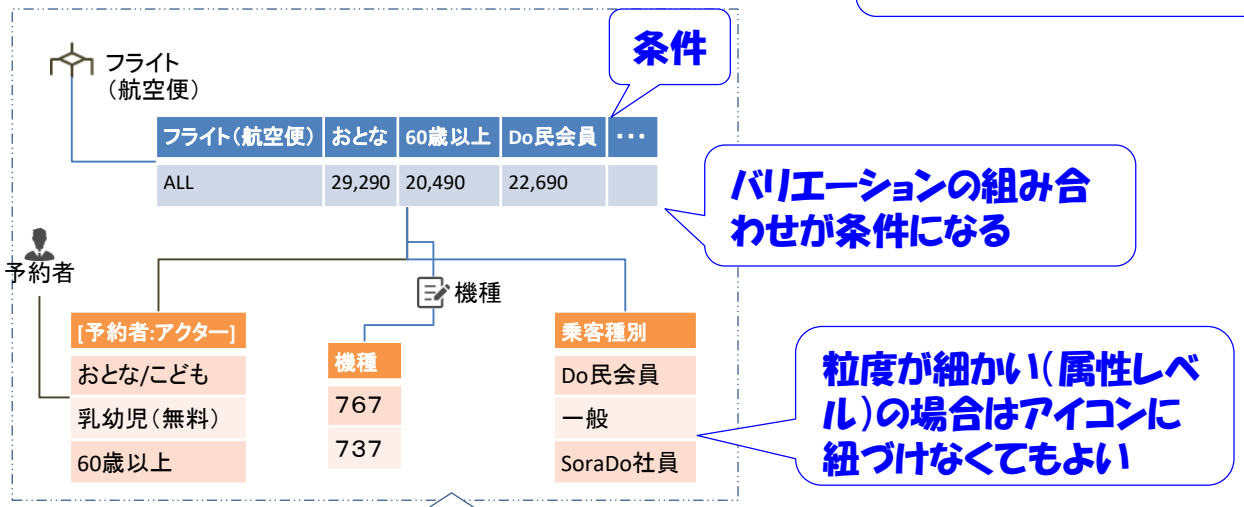


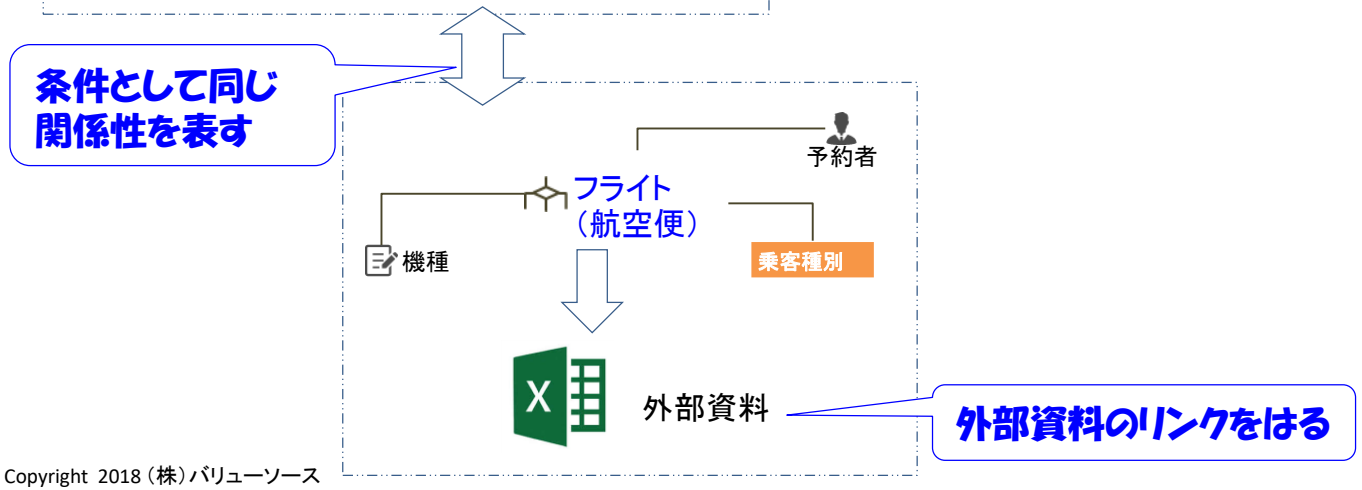
Copyright 2018 (株)バリューソース

ビジネスルール　条件・バリエーション





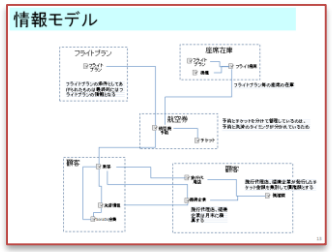




- このダイアグラムは何を表現するのか
  - ビジネスルールを表現
    - 条件
      - バリエーションの組み合わせで表現
        - » バリエーションを接続する
      - 条件アイコンをユースケースにつなぐ
    - バリエーション
      - 具体的な値を抽出
      - 保守開発において変更起点になる
    - Excelなどの外部資料は条件をリンクにする

Copyright 2018（株）バリューソース

情報モデル



座席在庫

フライトプラン

機種

フライト座席

フライトプラン毎の座席の在庫

フライトプラン

航空券

航空機予約

チケット

予約とチケットを分けて管理しているのは、
予約と決済のタイミングが分かれているため

B2C顧客

乗客

決済情報

SoraDo会員

B2B顧客

旅行代理店

旅行代理店、提携企業が発行したチケット金額を集計して債権額とする

債権額

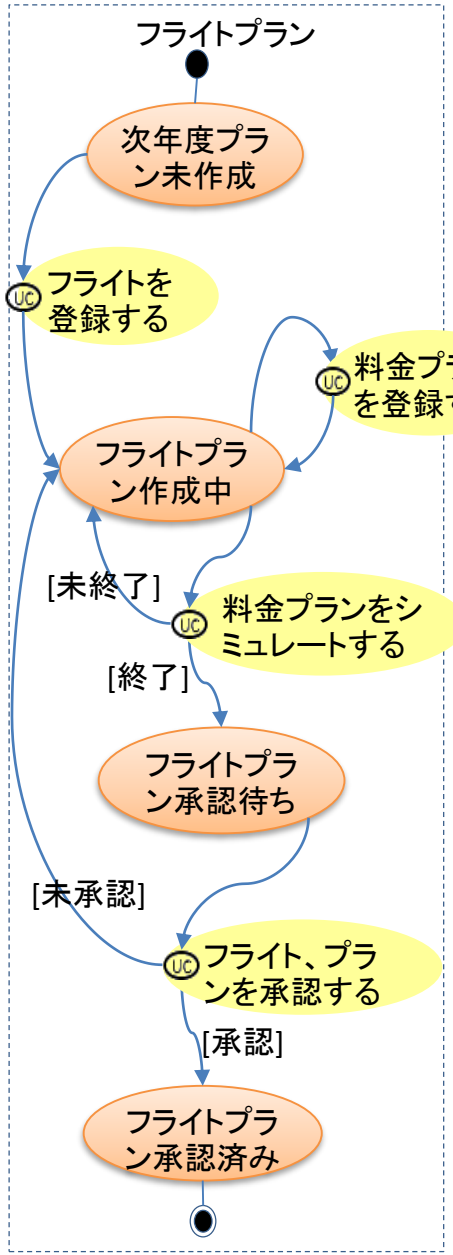
提携企業

旅行代理店、提携企業は月末に精算する

- 情報モデルは何を表すのか
  - システムを使って実現する情報をビジネス上の概念（ことば）を使って表現する
    - 概念モデルに近い
    - ドメインモデルではない
    - データモデルではない

Copyright 2018（株）バリューソース

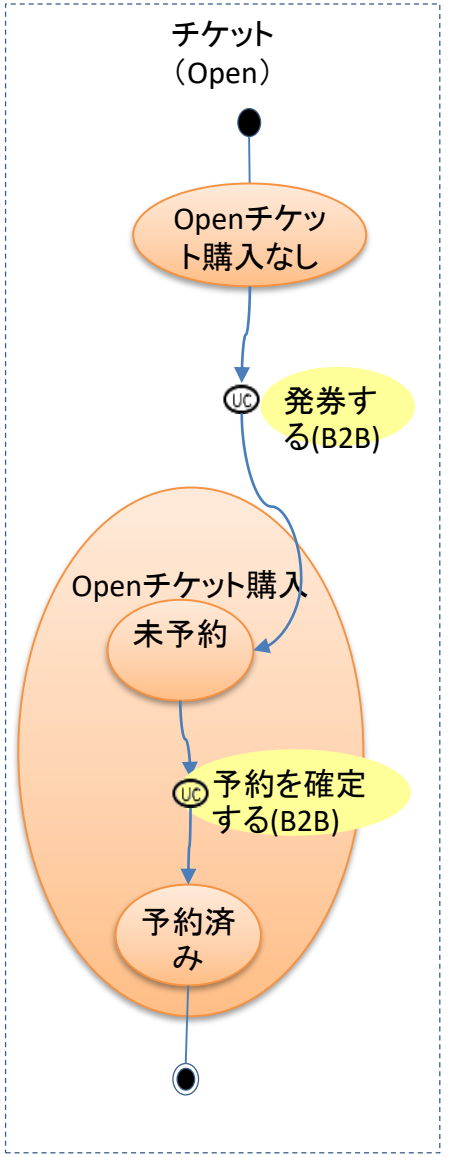
状態モデル



	Sora特28	Sora特21	Do直前X
SoraDo 101 08:00 - 09:35	¥10,690	¥14,290	¥15,490
SoraDo 103 09:00 - 10:35	¥11,790	¥15,590	¥16,890
SoraDo 105 10:20 - 12:00	¥10,690	¥14,290	¥15,490

フライト登録担当者

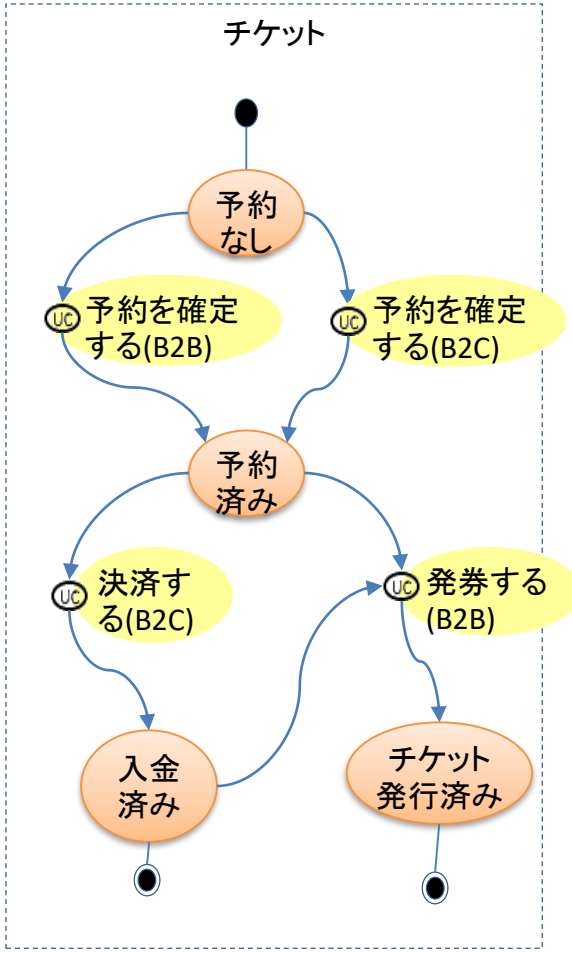
フライト登録管理者



状態モデルは何を表すのか

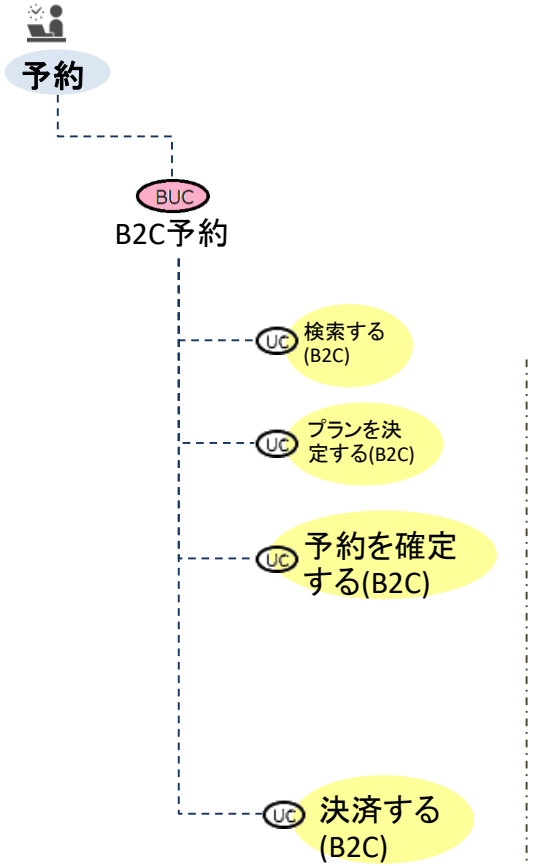
- システムが扱う状態をビジネス上の言葉で関係を表現
- 状態は情報で表現する
- 遷移は入出力に対応させる

	Sora特28	Sora特21	Do直前X
SoraDo 101 08:00 - 09:35	¥10,690	¥14,290	¥15,490
SoraDo 103 09:00 - 10:35	¥11,790	¥15,590	¥16,890
SoraDo 105 10:20 - 12:00	¥10,690	¥14,290	¥15,490

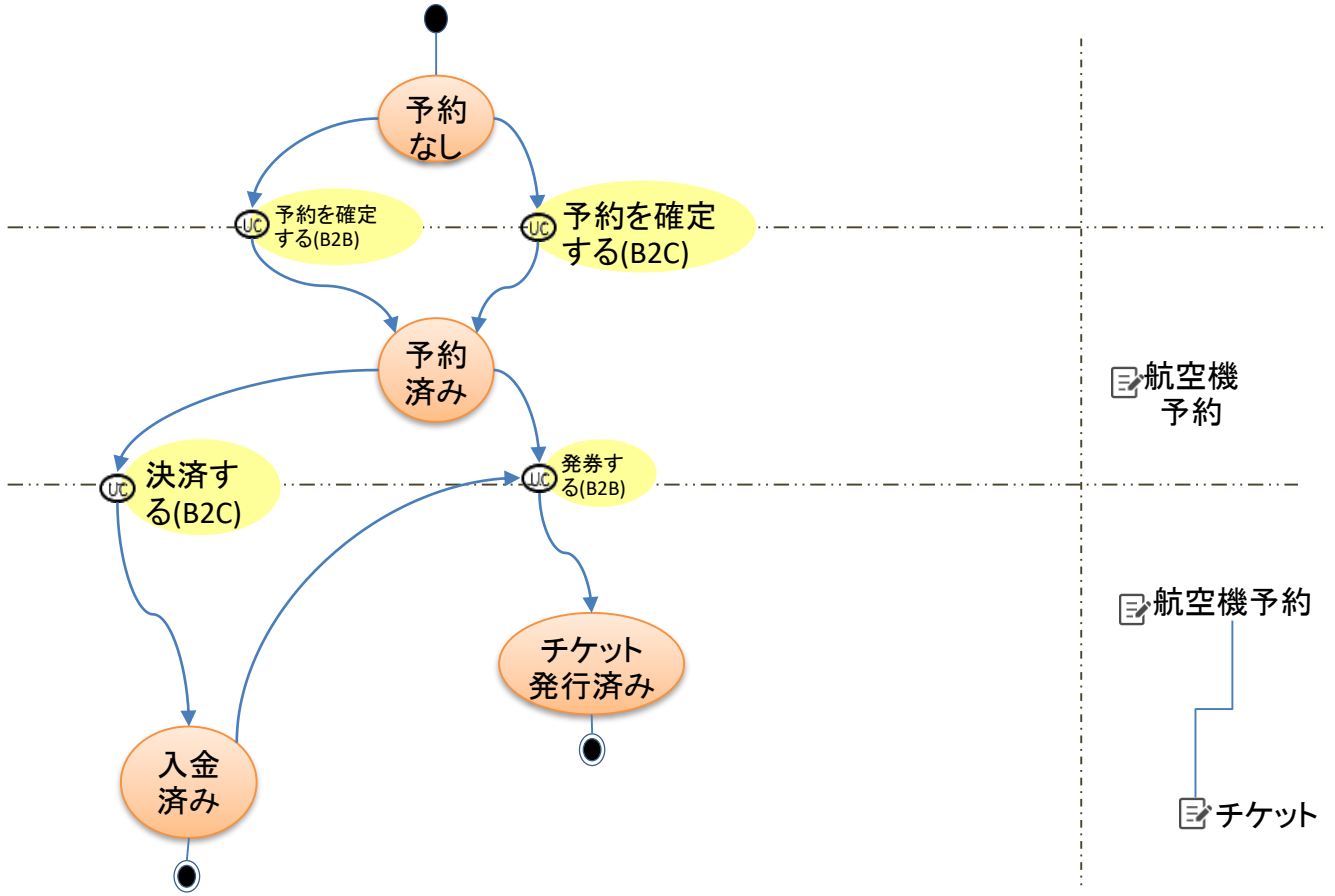


Copyright 2018 (株)バリューソース

ユースケースと状態と情報



- 状態は情報で表す
- 情報を変えるのがユースケース（入出力）
- 状態を変えるのがユースケース



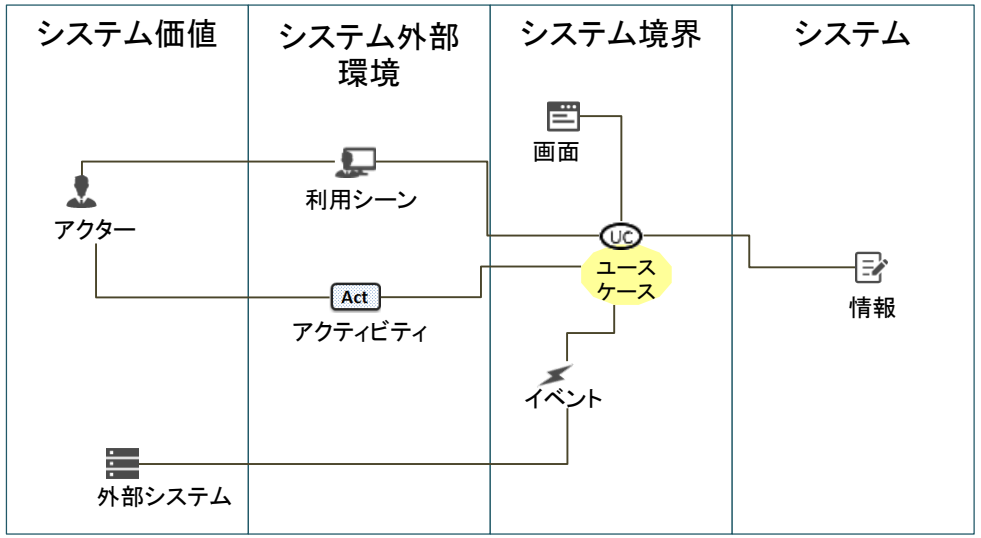
Copyright 2018 (株)バリューソース

システムを表現するための考え方

Copyright 2018 (株)バリューソース

システムの可視化要素

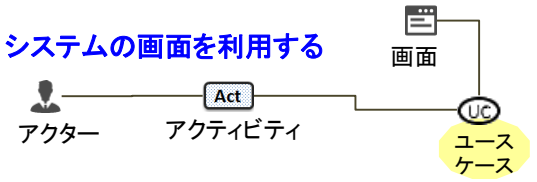
PowerPointを使った場合の可視化要素



多様な形態をアイコンの
つなげ方で表現する

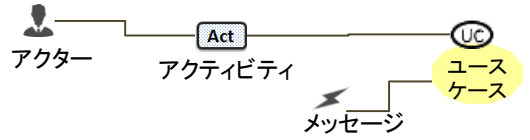
外部システムの画面を利用する

システムの画面を利用する

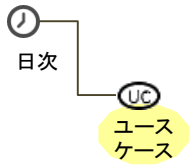


外部システムからのイベントで処理する

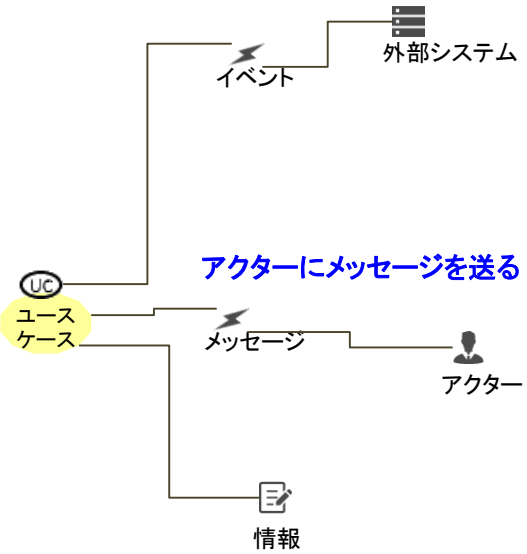
アクターからのメッセージングで処理する



定周期で処理する



外部システムと連携する

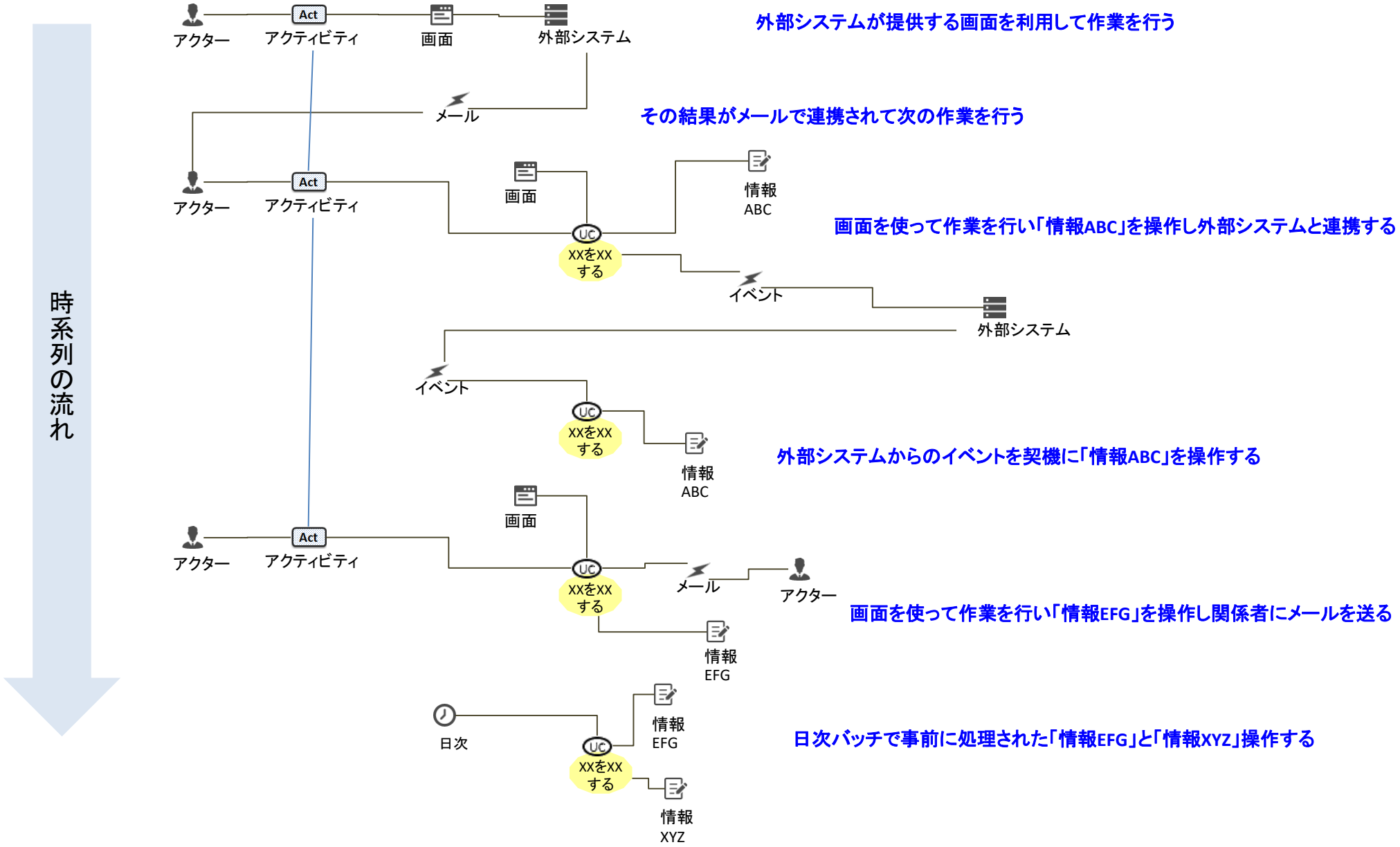


アクターにメッセージを送る

情報を操作する

Copyright 2018 (株)バリューソース

業務の流れとシステムとの連携を表現する



Copyright 2018 (株)バリューソース

変更起点を捉える

＜ビジネスの変更起点＞
手順とルール

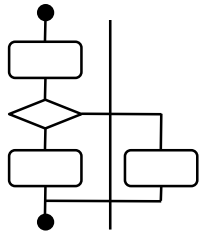
＜システムの変更起点＞
システムの入出力

ビジネスの条件が変わる

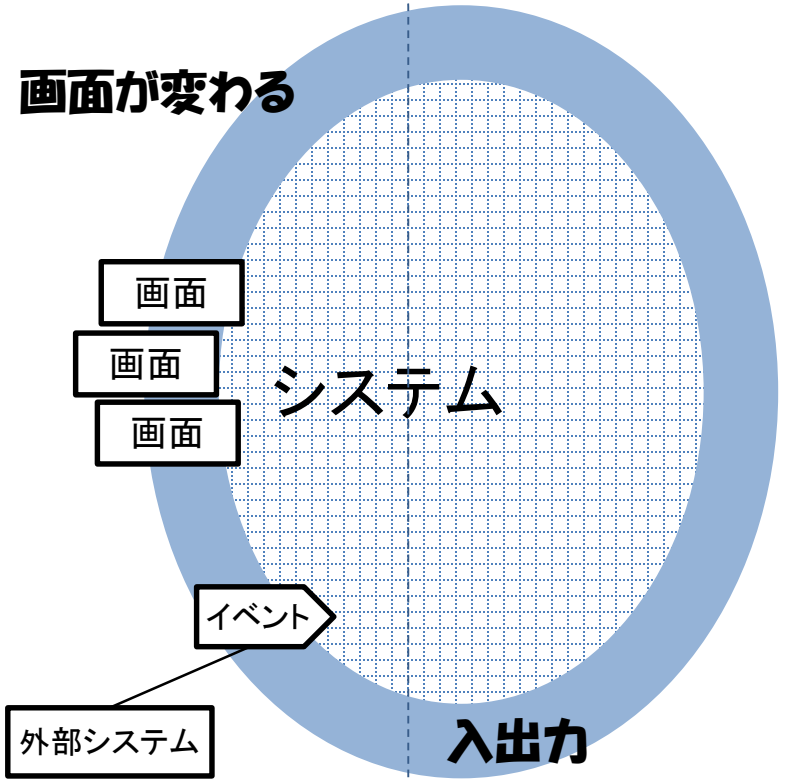
[得意先:情報]
大手流通
コンビニ
百貨店
卸

掛け率	コンビニ	大手小売り	卸
ランクA	0.97	0.96	1.0
ランクB	0.96	0.95	0.99
ランクC	0.95	0.95	0.98

仕事の流れが変わる



画面が変わる



連携方式が変わる

- 変更起点
  - ビジネス上の変更
    - 条件の追加・変更
    - 仕事の手順の変更
  - 入出力の変更
    - 画面の項目の追加変更
    - 外部システムとの連携方式、項目の変更

- 可視化のポイント
  - 変更起点があらかじめ明示されていれば、システム変更への対処が早くなる
    - ↓↓↓
  - 変更起点を捉えるようにまとめる

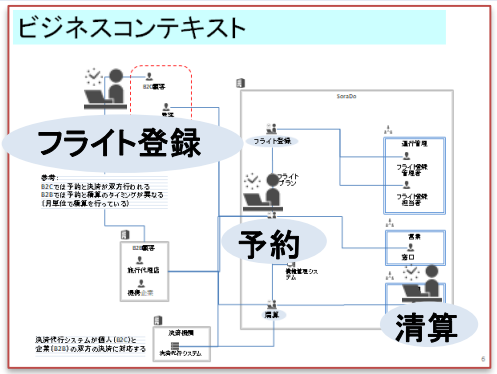
Copyright 2018 (株)バリューソース

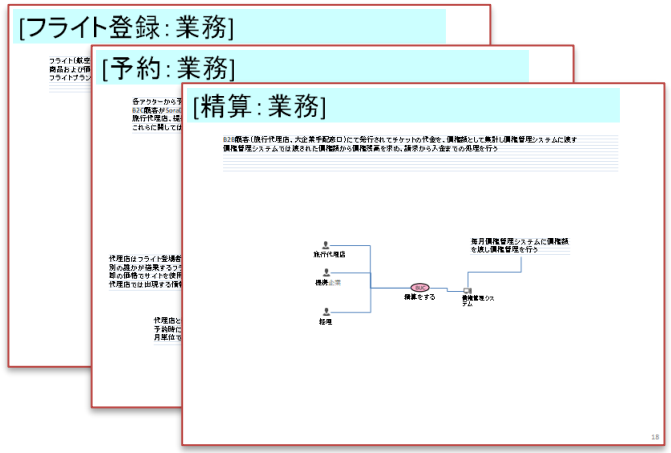
- 短期間でボリュームを捉えられるようにする
- 合意を得る仕組みを持ち関係者で共有する
- 類推できる粒度でメンテナンス可能なボリュームに保つ
- ビジネスルールで複雑度を測れるようにする
- 保守開発の影響度を測れるようにする
- 精度を向上させる仕組みをもつ

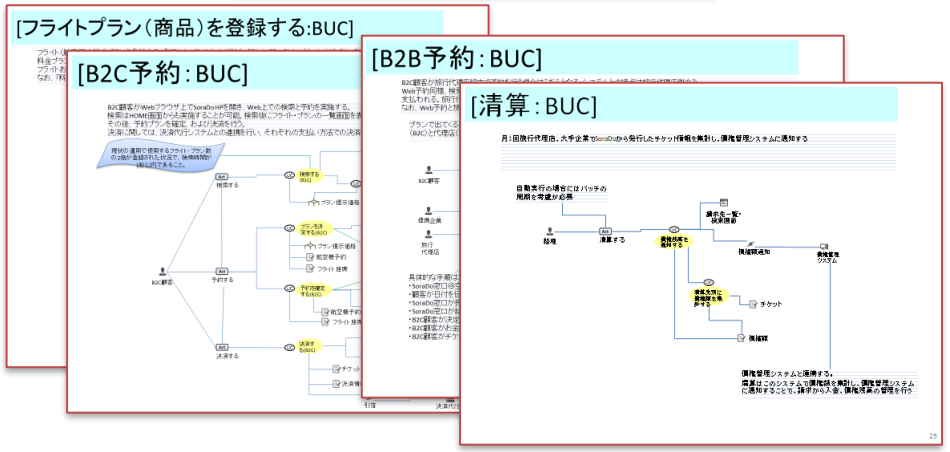
RDRAで問題を解決する

Copyright 2018 (株)バリューソース

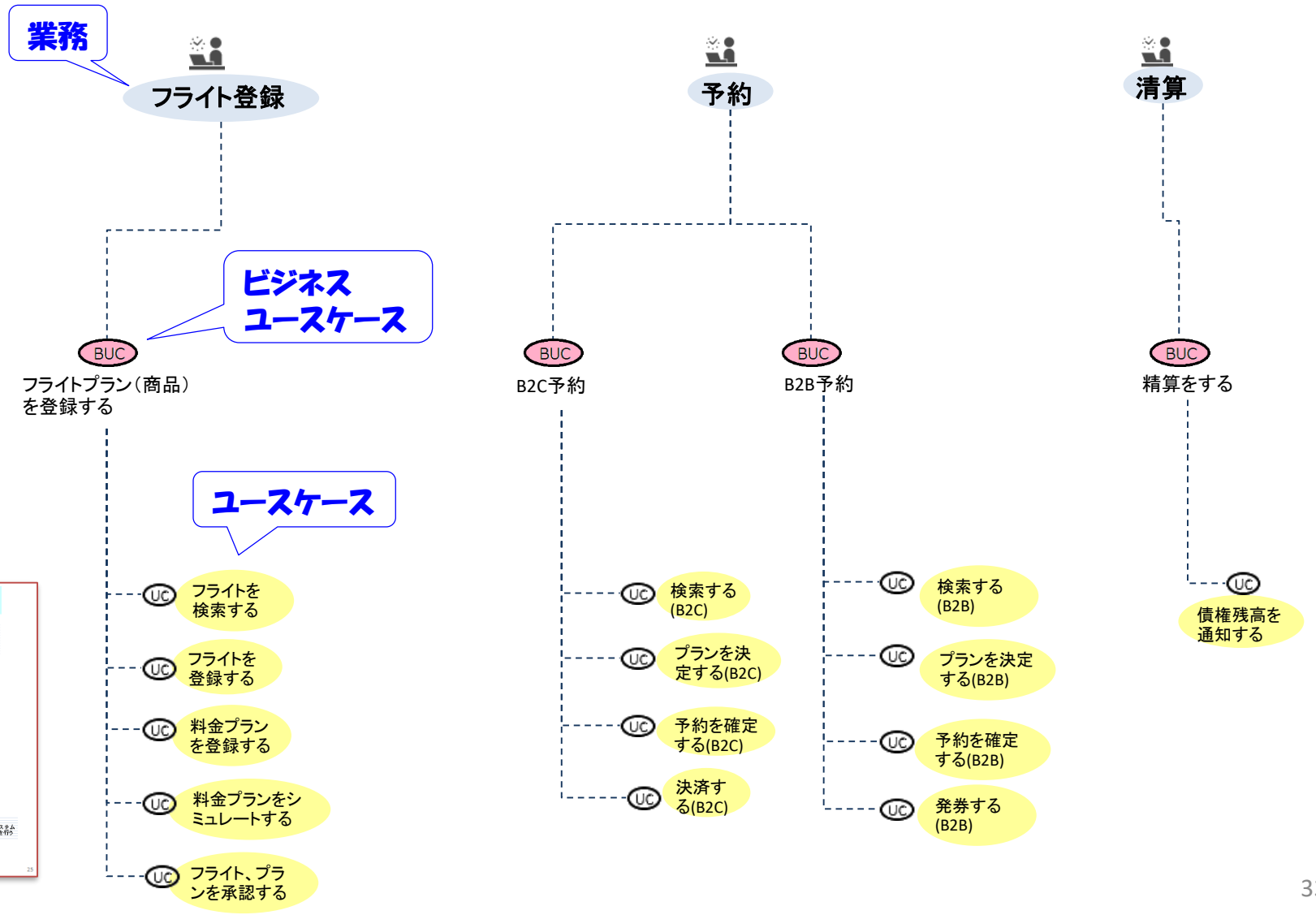
短期間でボリュームを捉えられる





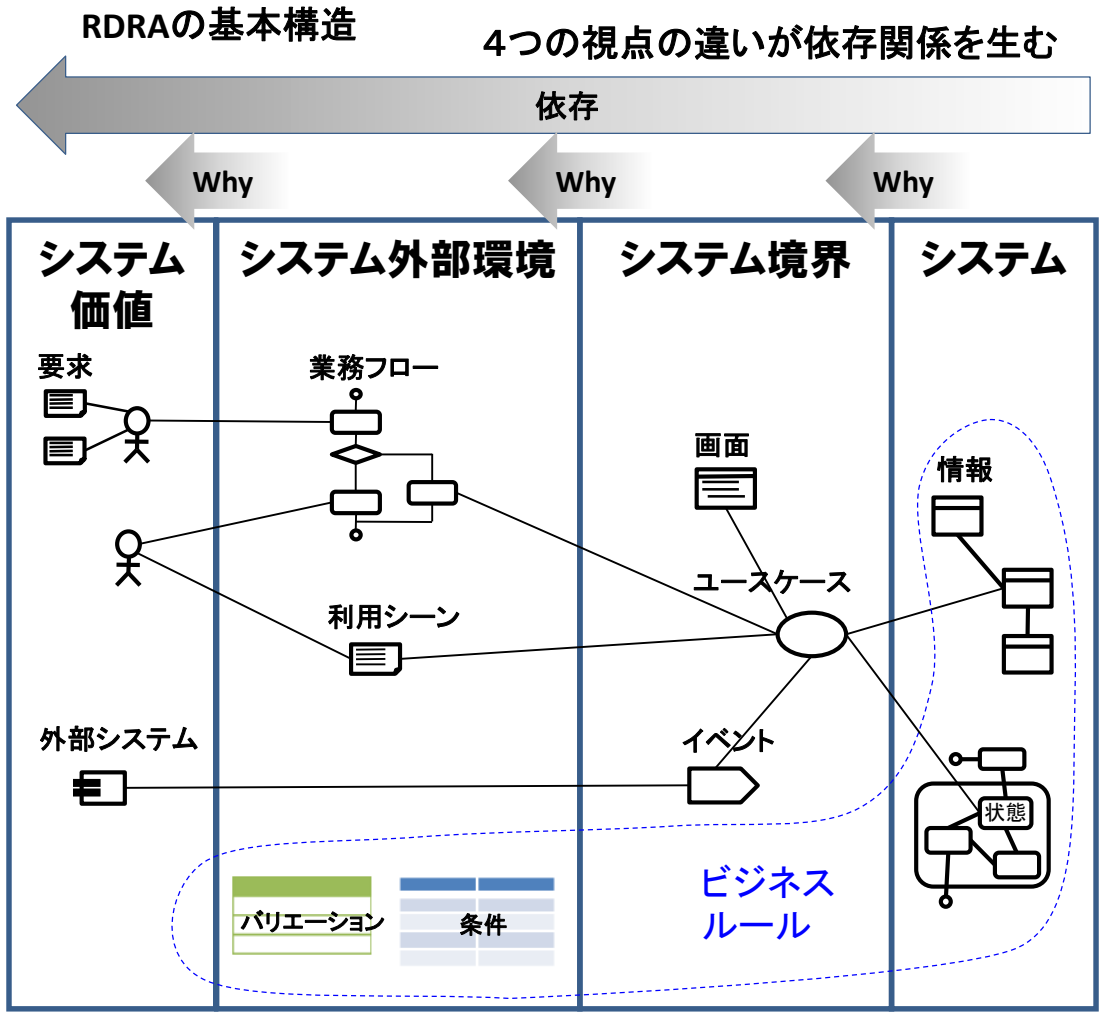


- 業務⇒ビジネスユースケース⇒ユースケースと階層的に詳細化しスコープを決める



Copyright 2018（株）バリューソース

合意を得るための仕組み

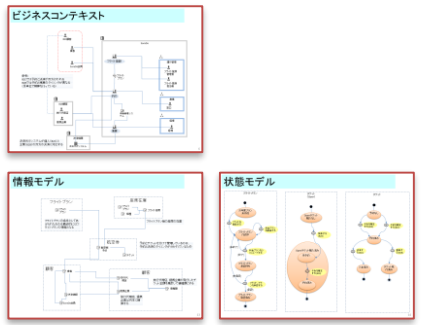


Copyright 2018 (株)バリューソース

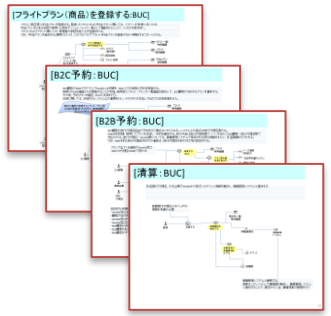
類推できる粒度でメンテナンス可能なボリュームに保つ

システム全体を表す仕様

現場の言葉で
表現する

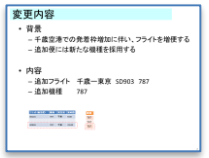


影響範囲を調
べて

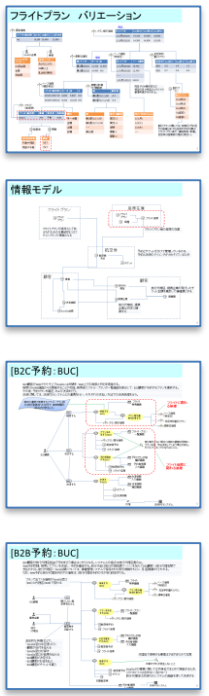


変更仕様

変更概要



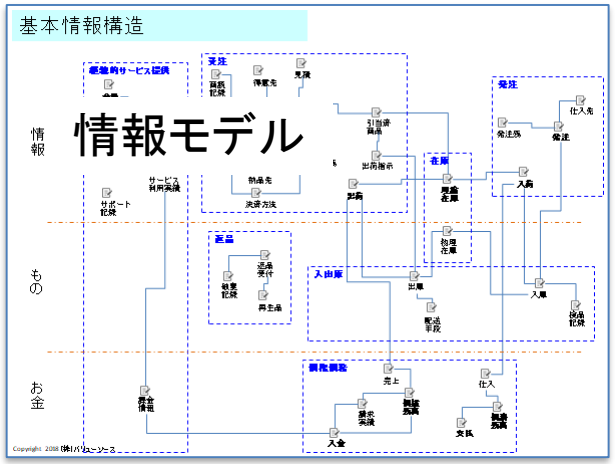
システム仕様の
変更ページをコ
ピーする



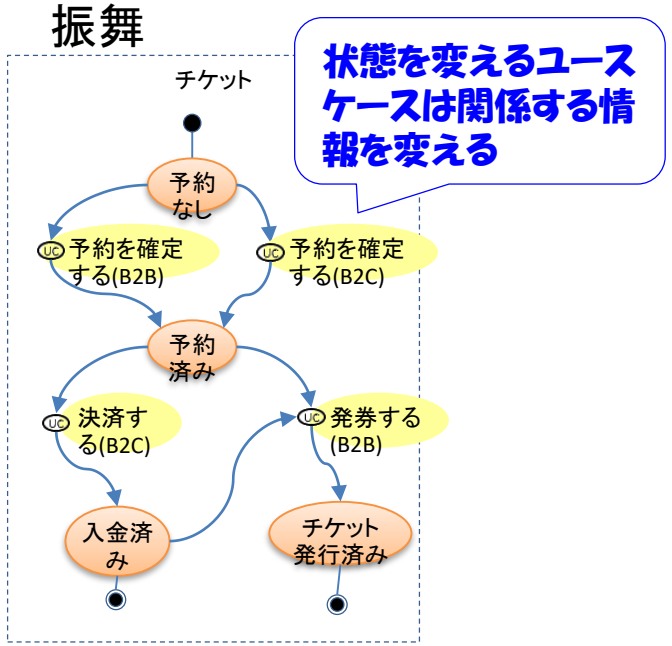
Copyright 2018（株）バリューソース

ビジネスルールで複雑度を測れるようにする

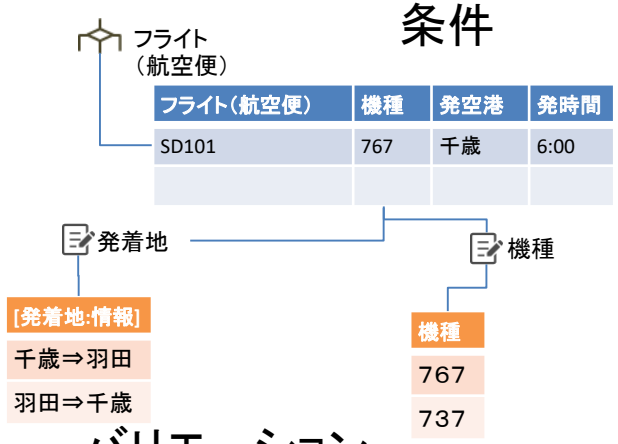
構造



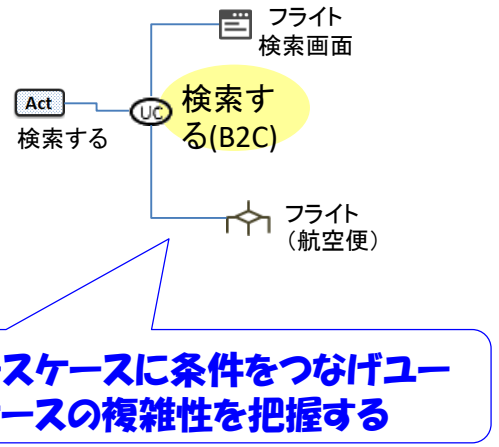
振舞



条件



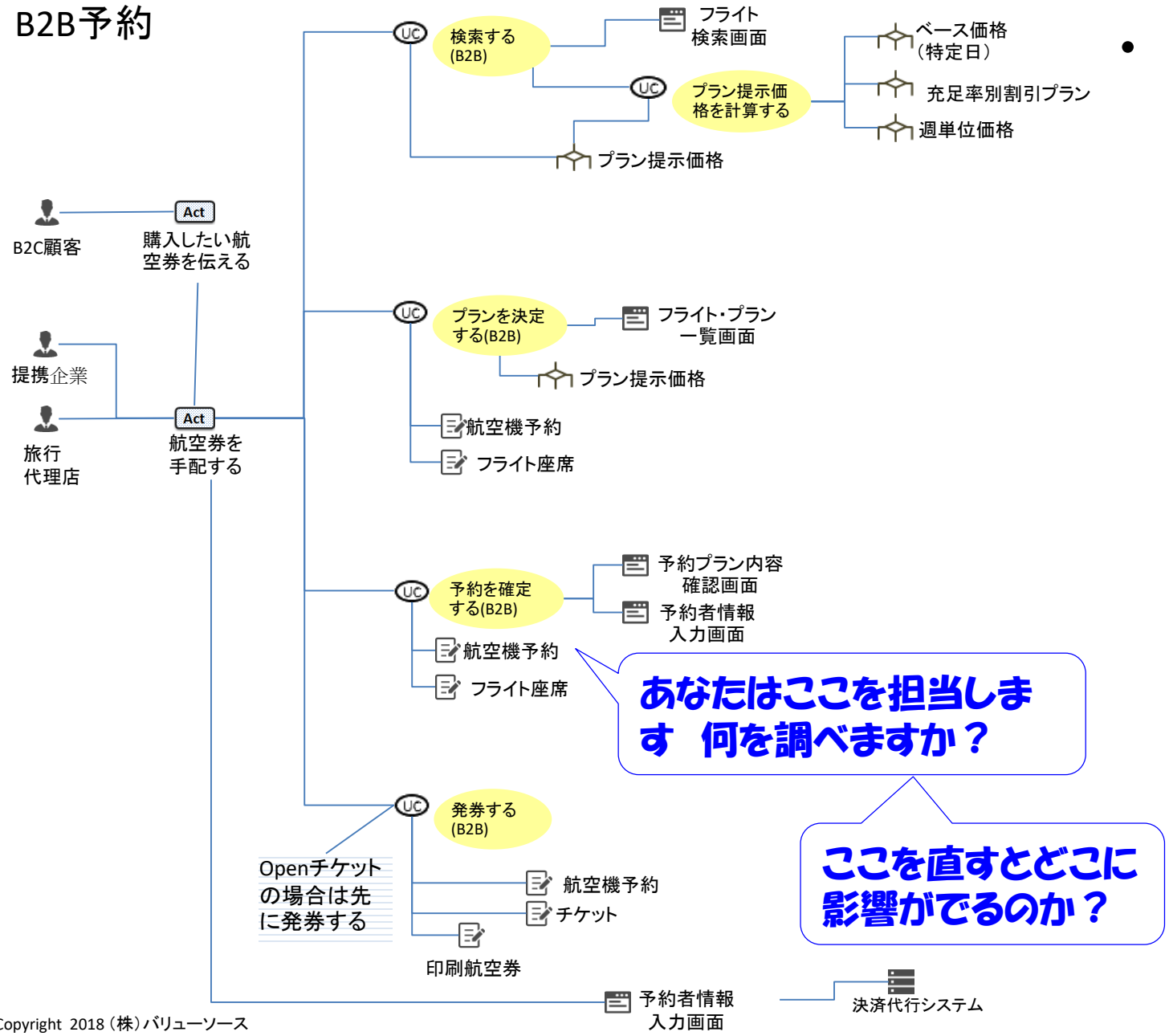
バリエーション



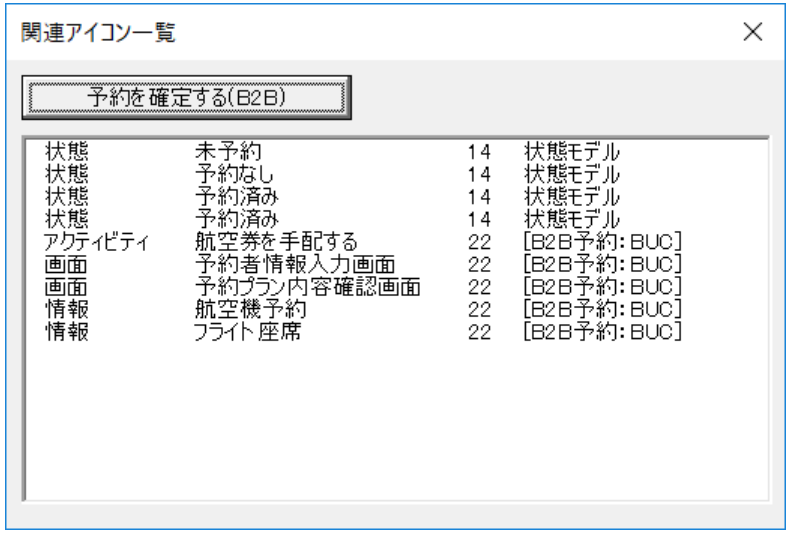
- ビジネスルールは3種類で表現できる
  - 構造
    - 情報モデル
  - 振舞
    - 状態遷移図
  - 条件・計算式 （表形式）
    - 計算式
    - 可能な組み合わせ
- 構造
  - 情報モデル：多重度
- 振舞
  - 状態遷移図：遷移可能な条件
- 条件・計算式
  - 条件
    - 条件はバリエーションの組み合わせとなる
  - 計算式
    - 金額＝数量＊単価
  - バリエーション
    - 条件の組み合わせとなるバリエーションを洗い出す

Copyright 2018 (株)バリューソース

保守開発の影響度を測れるようにする

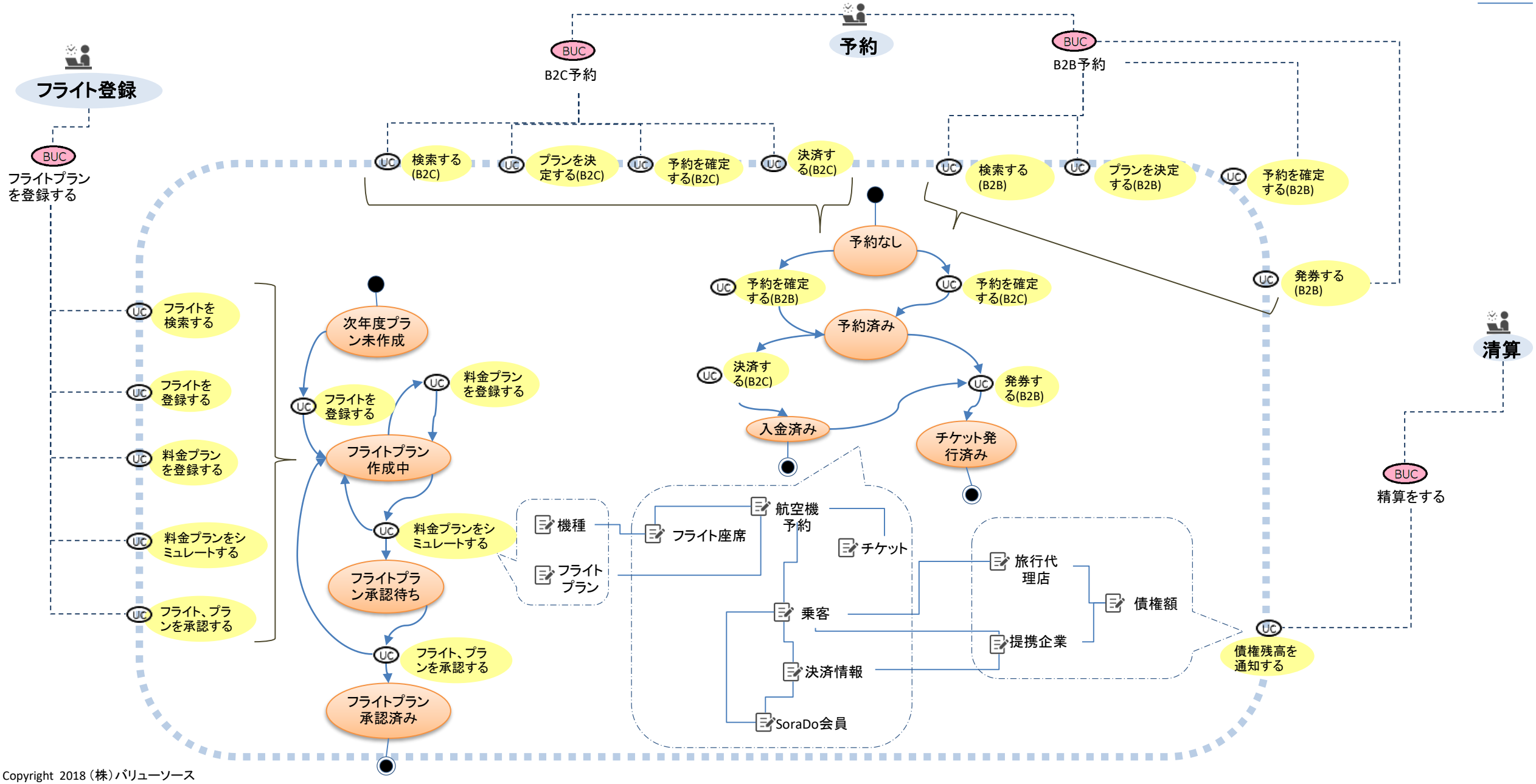


- 「予約を確定する(B2B)」を担当します
  - 何を気にしますか
    - 他で使われていないか
      - 「航空機予約」
      - 「予約プラン内容確認画面」
      - 「予約者情報入力画面」
      - 「予約を確定する(B2B)」



Copyright 2018（株）バリューソース

精度を向上させる仕組みをもつ



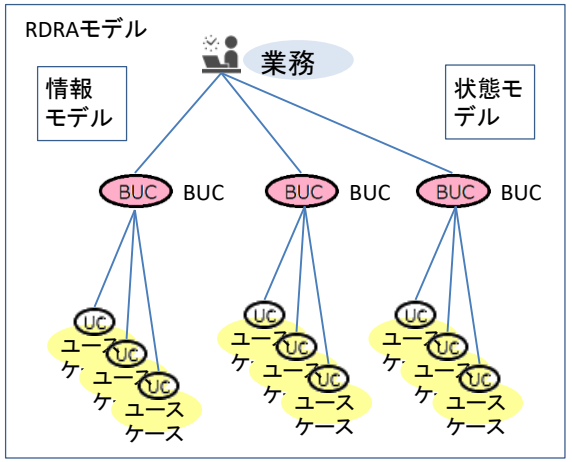
Copyright 2018 (株)バリューソース

まとめ

Copyright 2018 (株)バリューソース

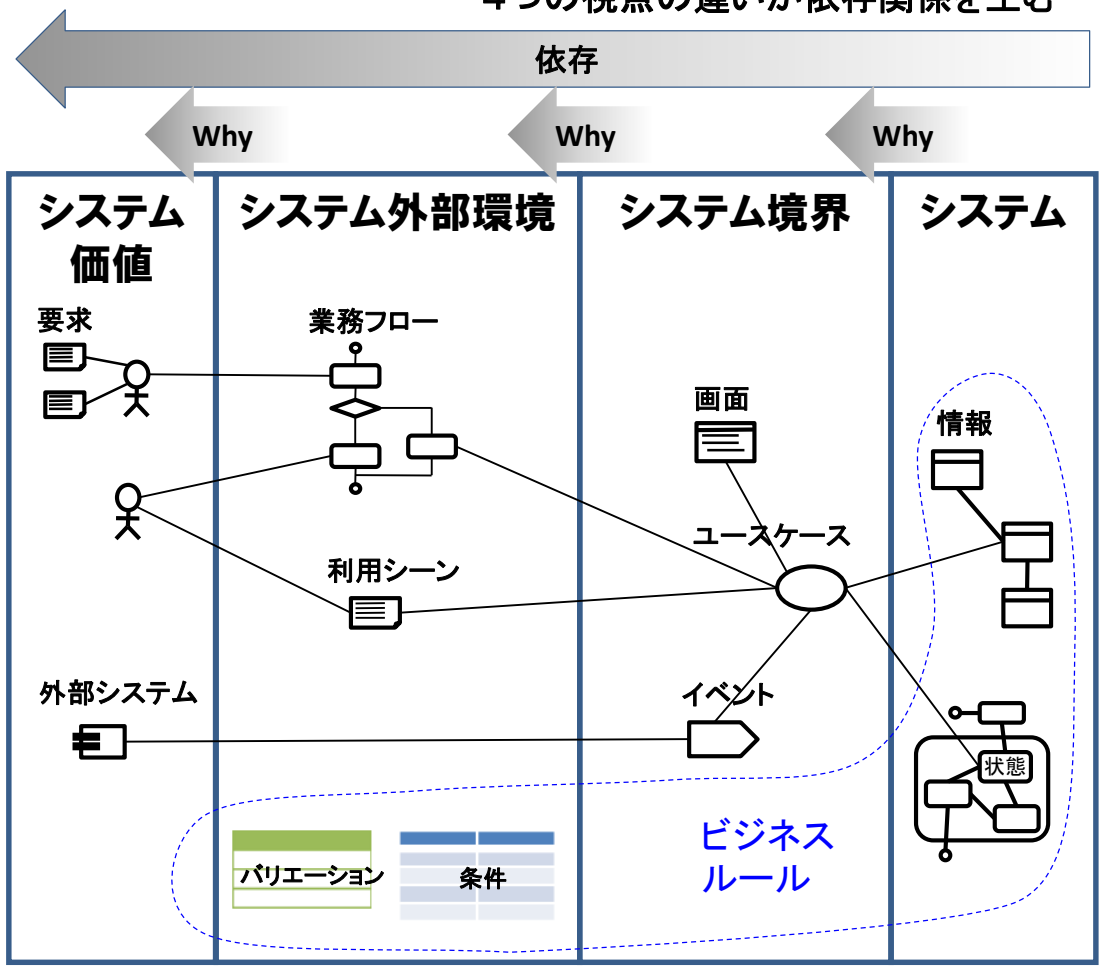
4つの視点で全ての情報をつなげて要件を定義する

階層化された文脈で体系化する



RDRAの基本構造

4つの視点の違いが依存関係を生む

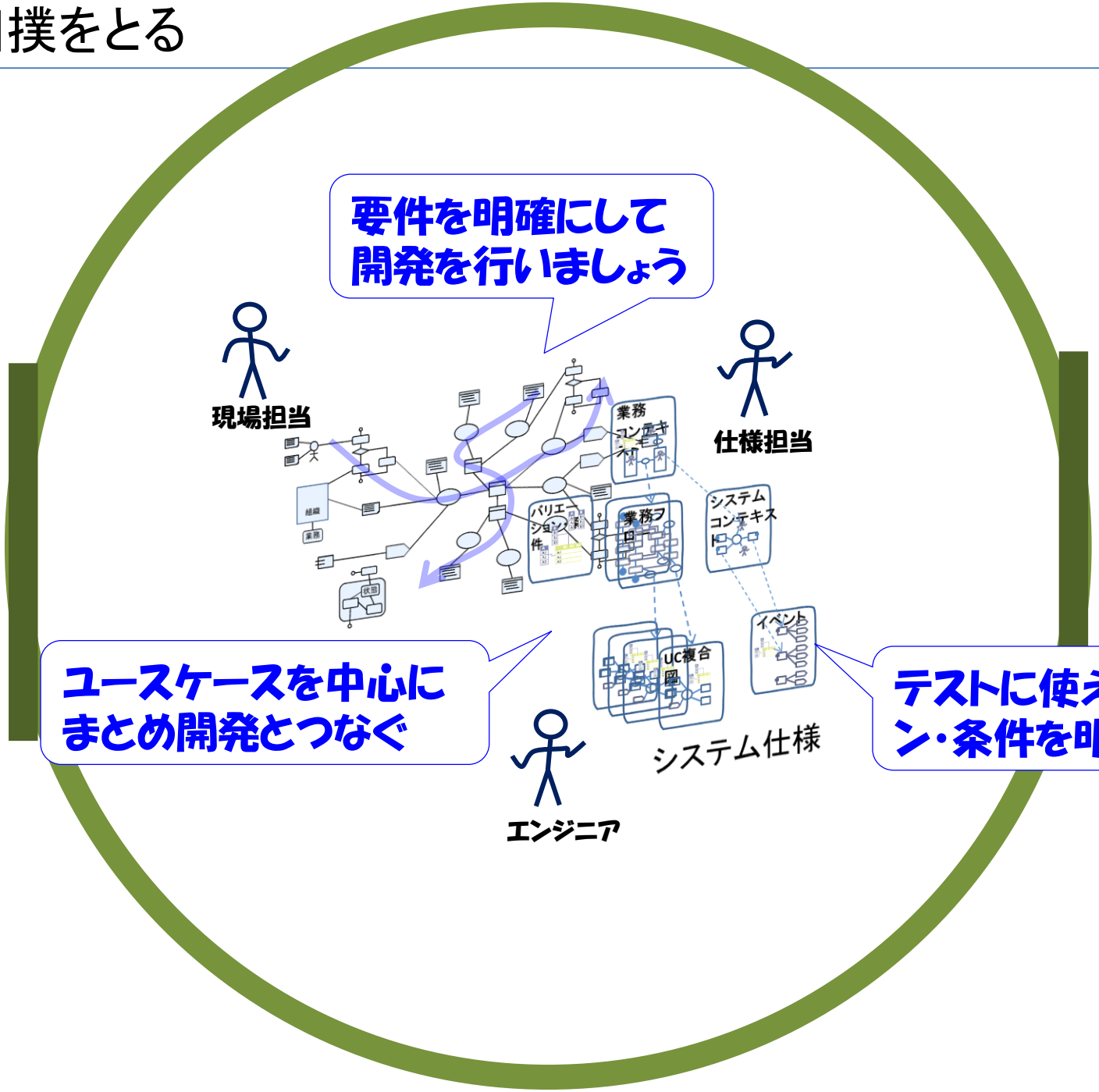


現場の「ことば」で表現する

- 短期間でボリュームを捉えられるようにする
  - ⇒ 素早くユースケースを導出できる仕組みを持つ
- 合意を得る仕組みを持ち関係者で共有する
  - ⇒ 要件の決め方を持つ
- 類推できる粒度でメンテナンス可能なボリュームに保つ
  - ⇒ 現場の「ことば」を使って要件化する
- ビジネスルールで複雑度を測れるようにする
  - ⇒ バリエーションの組み合わせでビジネスルールを表現する
- 保守開発の影響度を測れるようにする
  - ⇒ アイコンをつなげて関係性を表す
- 精度を向上させる仕組みをもつ
  - ⇒ 情報と状態、ユースケースの関係で整合をとる

Copyright 2018 (株)バリューソース

土俵の真ん中で相撲をとる



Copyright 2018 (株)バリューソース