



オブジェクト倶楽部2008夏イベント

ビジネスアプリケーションによるモデル駆動設計、
そしてプロジェクトについて想うこと

国立オリンピック記念青少年総合センター

2008年7月1日 14:50 – 16:30

株式会社 シナジー研究所

依田 智夫 (yoda@synergy-res.co.jp)

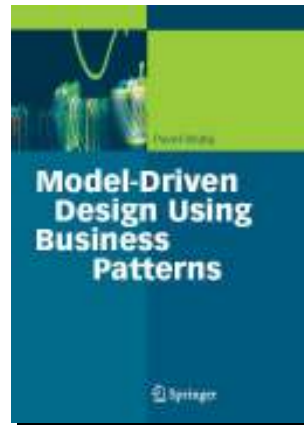


自己紹介

- 株式会社 シナジー研究所
 - 代表取締役 プリンシパル・コンサルタント
- 総務省 行政管理局 技術顧問
- 要求開発アライアンス理事
 - <http://www.openthology.org>
- 共著書：
 - 「要求開発 – 価値ある要求を導き出すプロセスとモデリング -」 (日経BP社、2006年)
- 監訳書：
 - 最終版のみ記載
 - 実践UML 第3版 オブジェクト指向分析設計と反復型開発入門 (ピアソン・エデュケーション、2007年)
 - 「ビジネスパターンによるモデル駆動設計」 (日経BPソフトプレス、2007年)
 - 以下、すべてピアソン・エデュケーション社
 - 「UMLによるXMLアプリケーションモデリング」 (2002年)
 - 「UMLによるWEBアプリケーション開発」 (2000年)
 - 「UMLによるJavaオブジェクト設計」 (2000年)
 - 「Javaエンタープライズ・コンポーネント」 (2000年)
 - 「戦略とパターンによるビジネスオブジェクトモデリング」 (1999年)

1. REAとビジネスパターンの概要
2. 質問コーナー
3. 構造パターンと振る舞いパターン
4. REAと開発プロジェクト
4. 簡単なREAモデリング演習

ビジネスパターンとモデル駆動設計



Pavel Hruby, Jesper Kiehn, Christian Vibe Scheller
Model-Driven Design Using Business Patterns,
Springer, 2006 (English edition)
Nikkei BP, 2007 (Japanese edition)

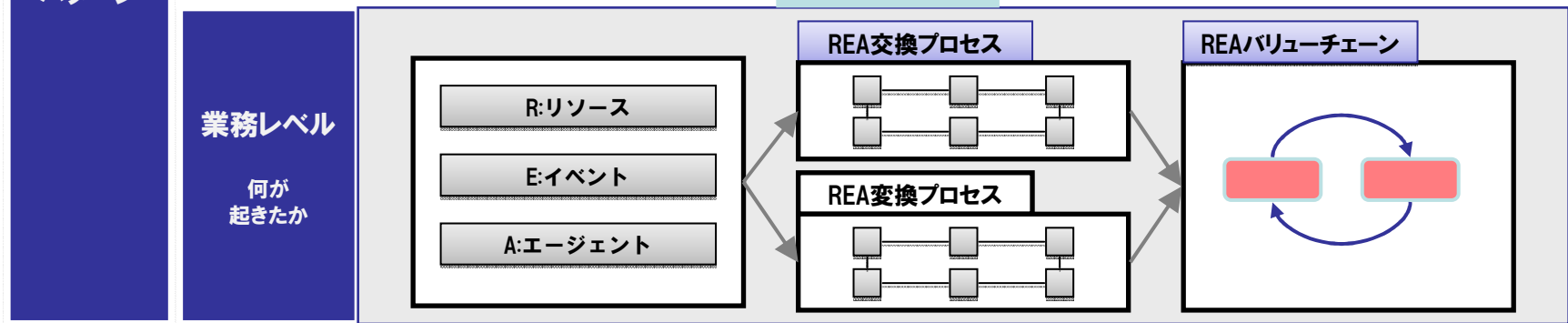
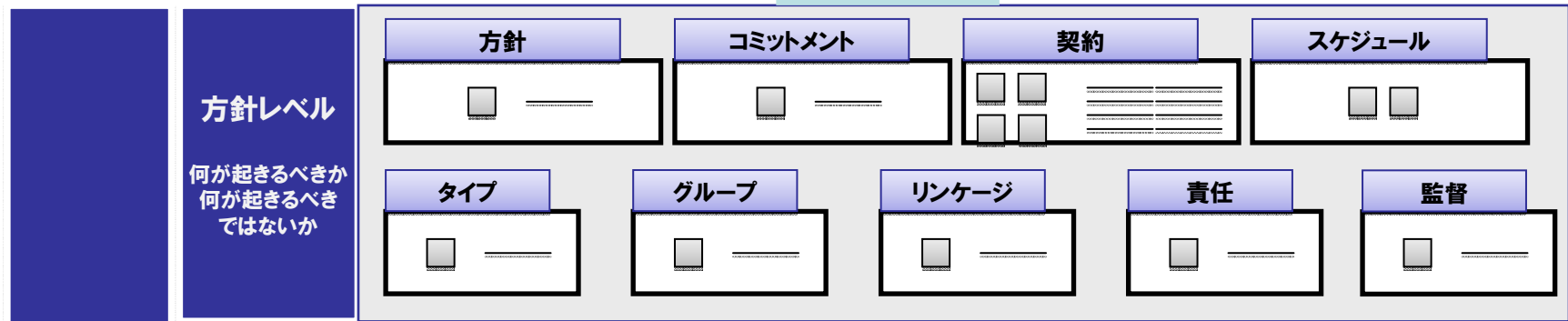
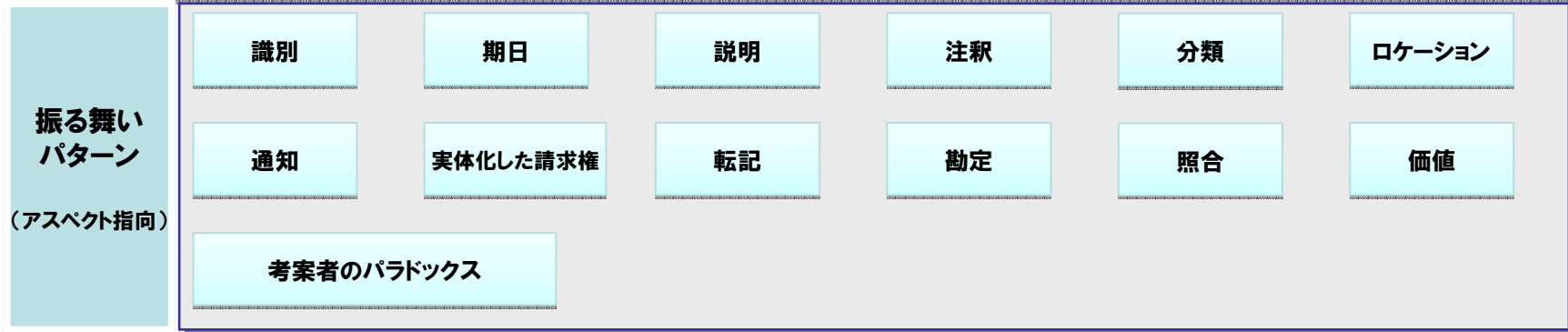
- 会計のためのデータモデルとしてミシガン州立大学のWilliam E. McCarthyによって1982年に提案された
- ビジネスプロセスによって現実世界の意味づけを行い、その意味に基づいて経済的な取引やそれに付随するデータの関連をモデル化することで、データ間のトレーサビリティを提供する
 - なぜそのような取引が起きたのか
 - その取引の結果はどうなるのか
- 米国の大学の企業情報システムの講座で教えられている
- すでに4冊の書籍が刊行
- 本年8月に日本で刊行された「ビジネスパターンによるモデル駆動設計(依田が監修を担当)」には、REAに基づく各種のパターンと、それらを利用してアプリケーションを開発する手法が具体的に紹介されている。
- Eコマース標準作成の基盤を提供している
 - ebXML
 - UN/CEFACT(貿易簡易化と電子ビジネスのための国連センター)、
 - Open-edi (ISO/IEC 15944-4)
 - オントロジーフレームワーク: Open-edi business transaction ontology (OeBTO)

著者Pavel Hruby氏の主張について依田のまとめ

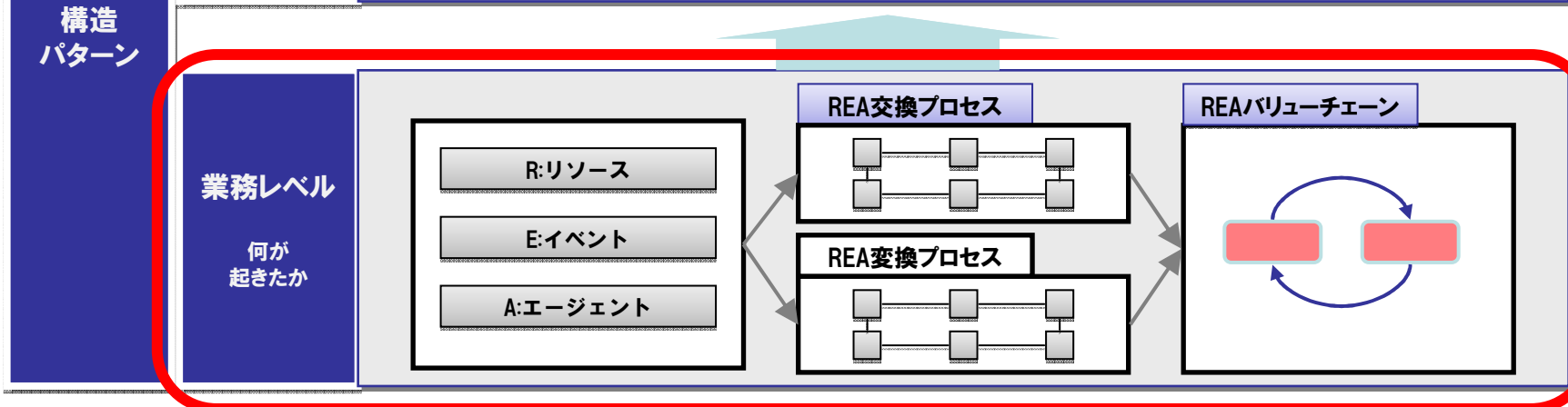
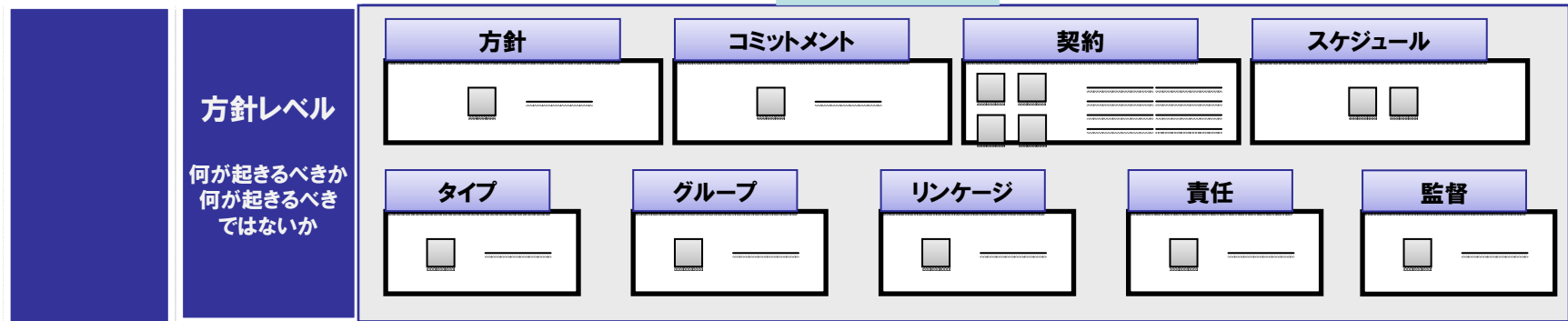
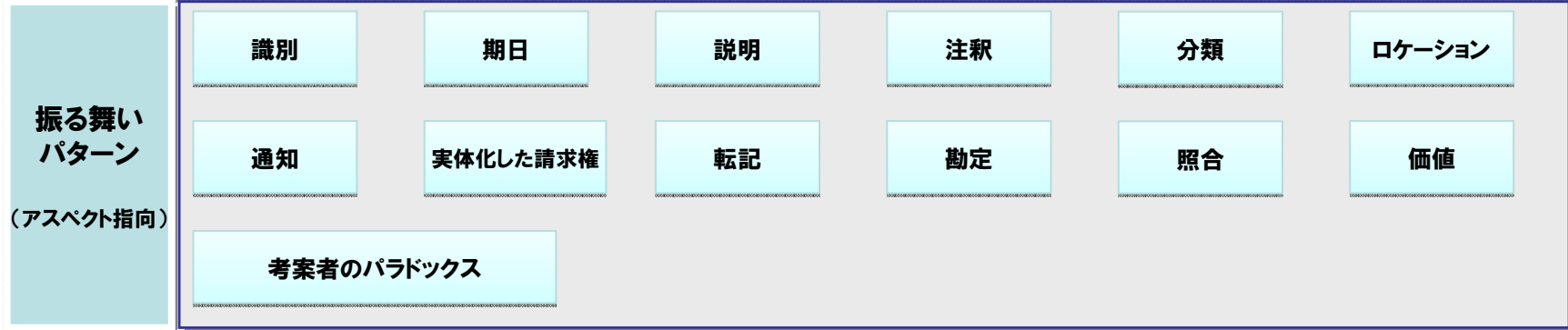
1. 企業が関係する資源の計画と管理に関する情報システムは、経済イベントの対称性に基づいてモデル化すると高品質なシステムが開発できる。
2. そのようにモデル化されたアプリケーションは、パターンを利用して高速に開発し効率的に保守することができる(REAパターン)。
3. そのようにモデル化され開発されたアプリケーションは会計システムの情報を包含し(財務諸表が出せる)、なおかつトレーサビリティが高く理解の容易な情報を提供するものとなる。
4. 中小企業もカスタマイズが容易で安価な独自のERPシステムを持つことができる。

「ビジネスパターンによるモデル駆動設計」は、1、2について書かれた本です。
本日のお話も1、2を中心に行います。
3、4については“Elevator Pitch, REA TECHNOLOGY”などを参考にしてください。

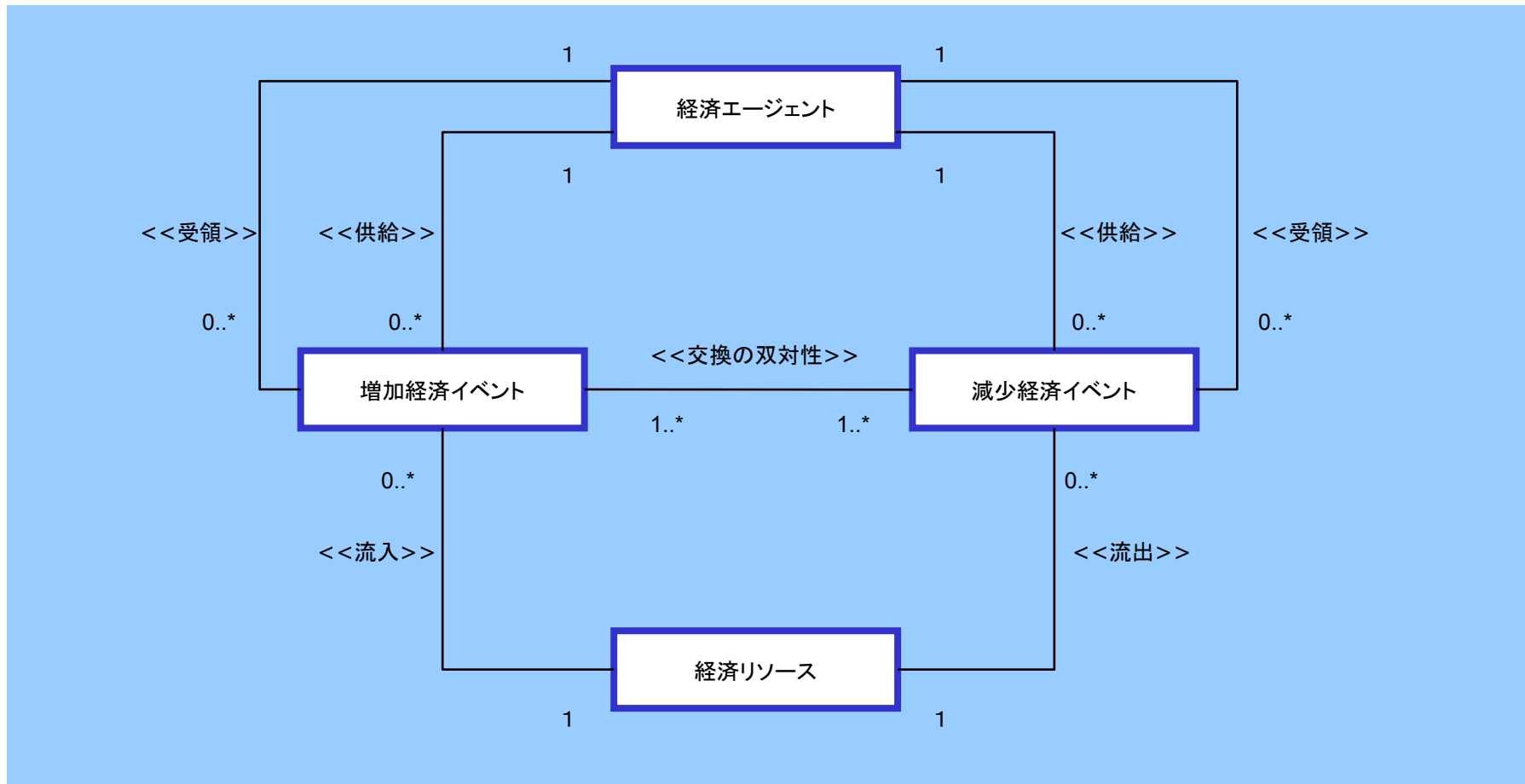
REAビジネスパターンの全貌



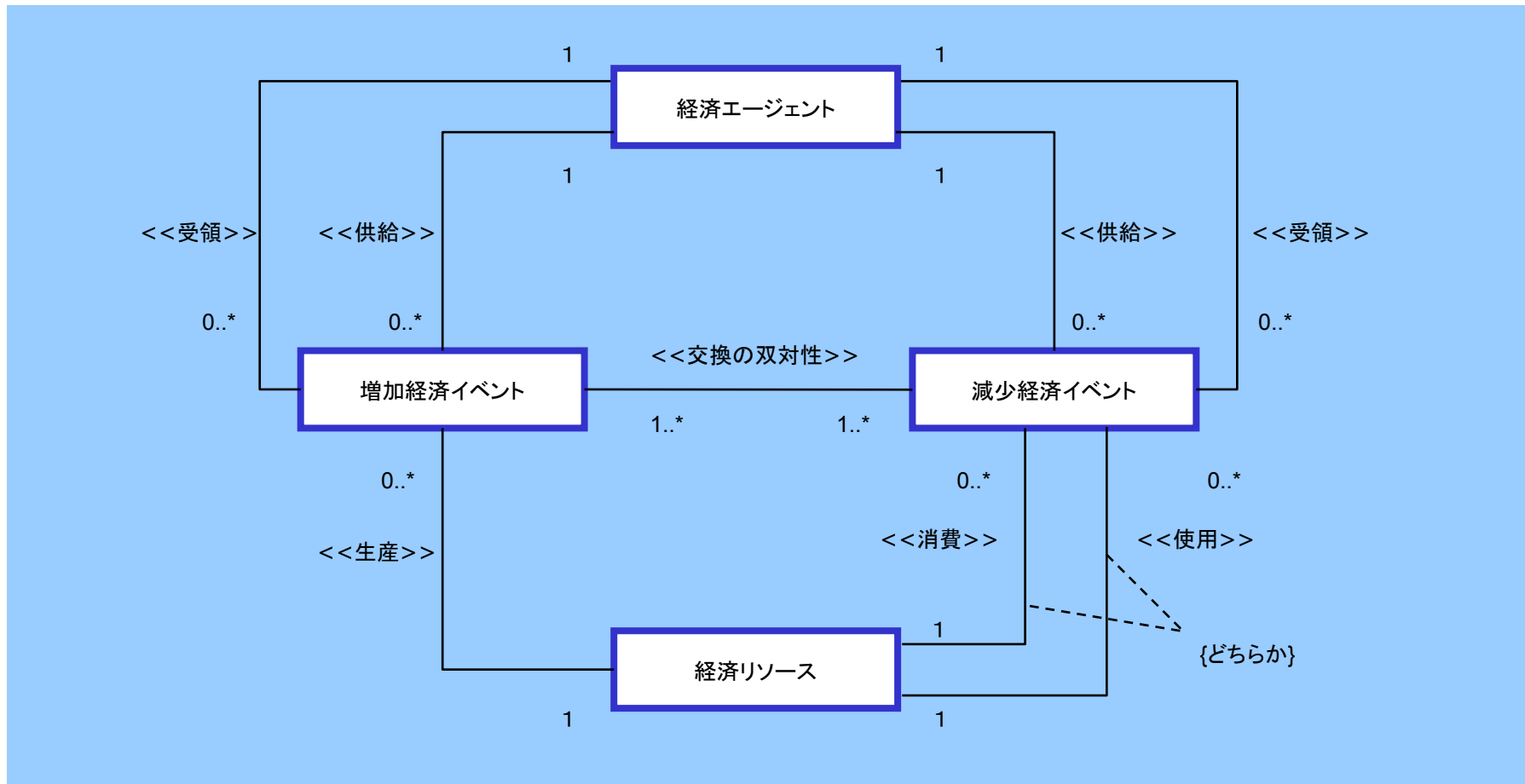
構造パターン(業務レベル)



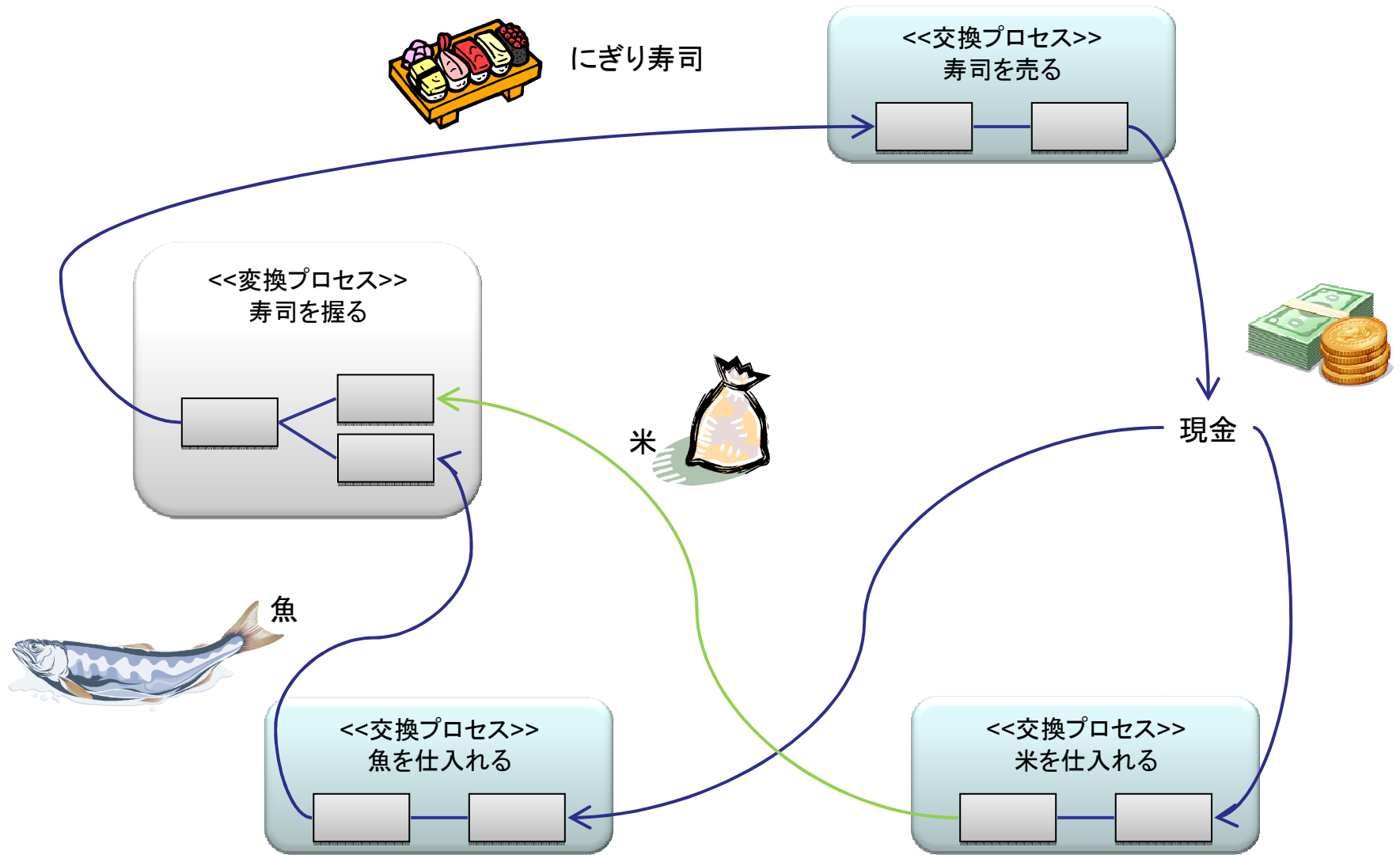
REA交換プロセスパターン



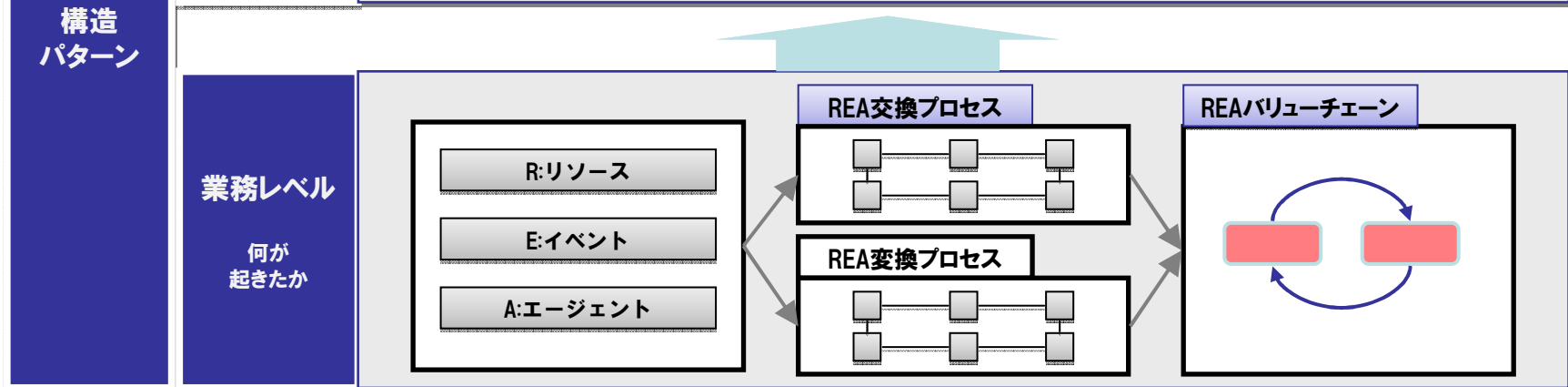
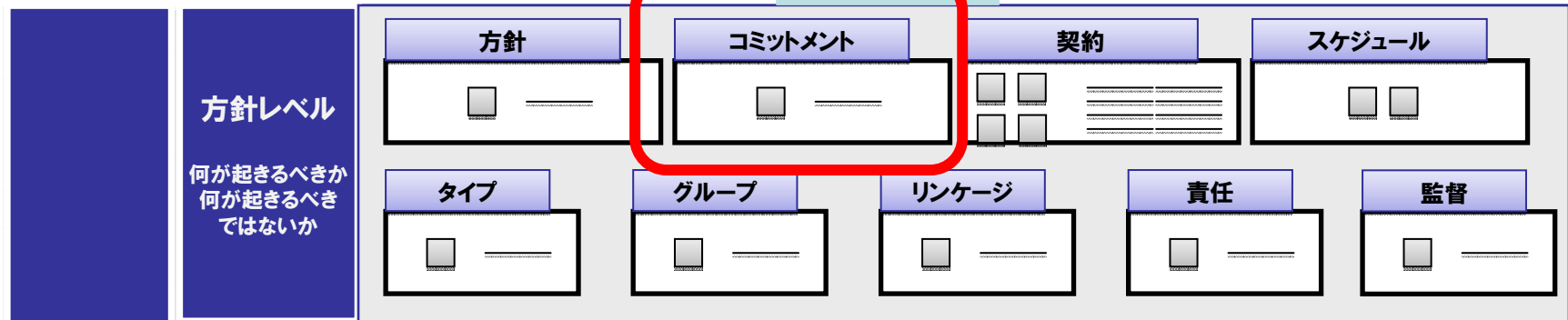
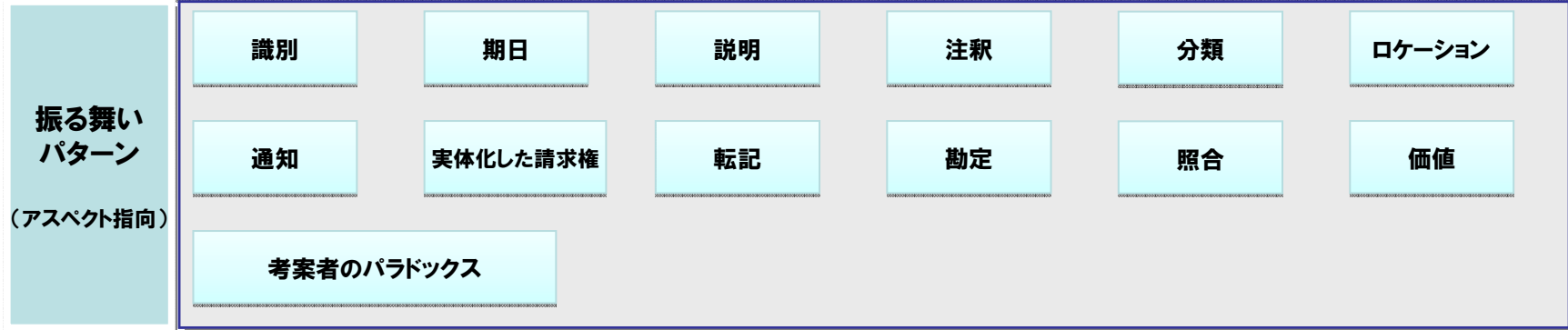
REA変換プロセスパターン



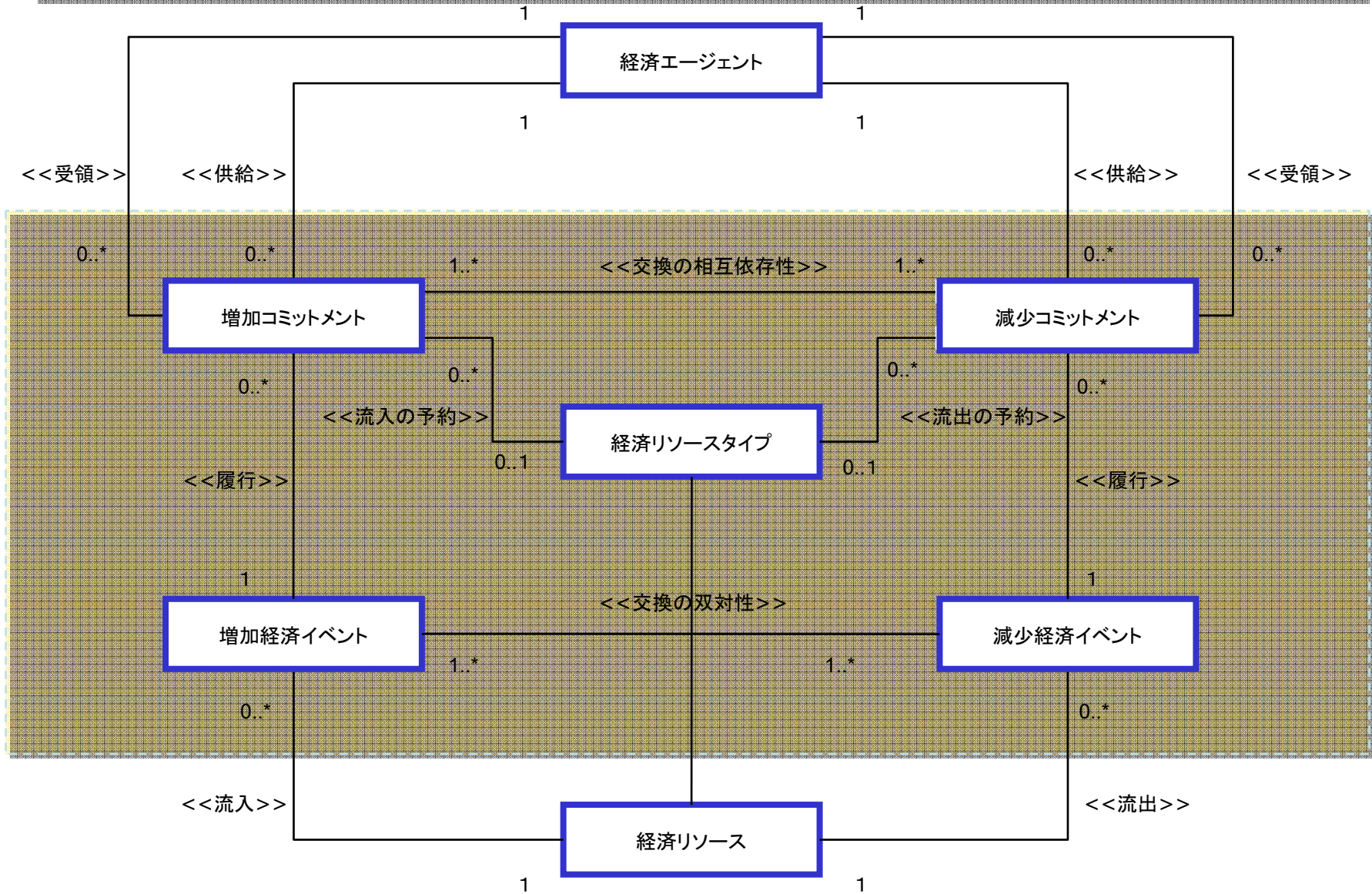
バリューチェーン・パターン



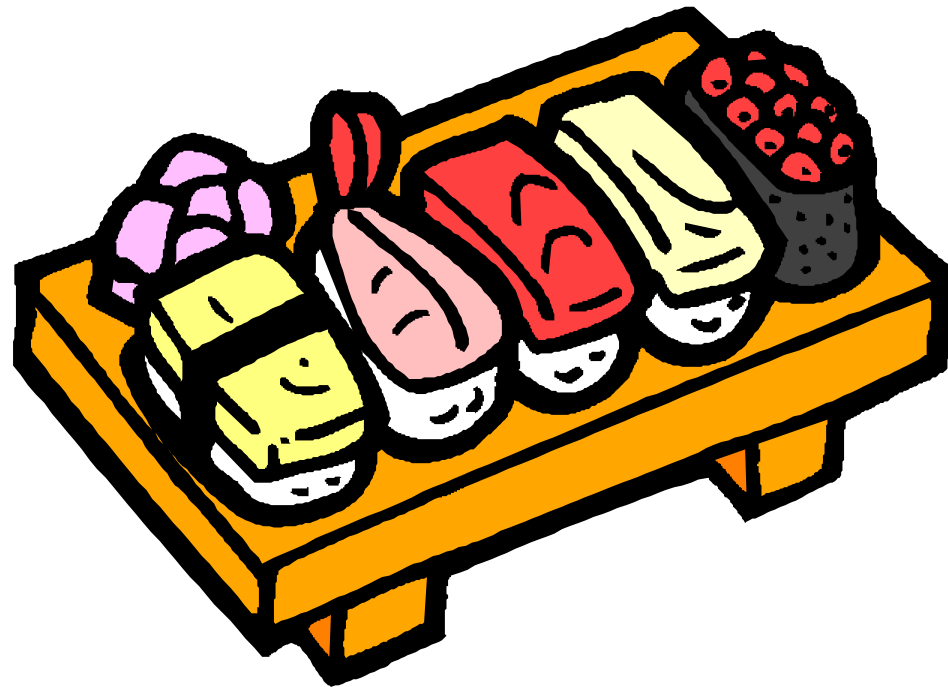
構造パターン(方針レベル)



コミットメントパターン

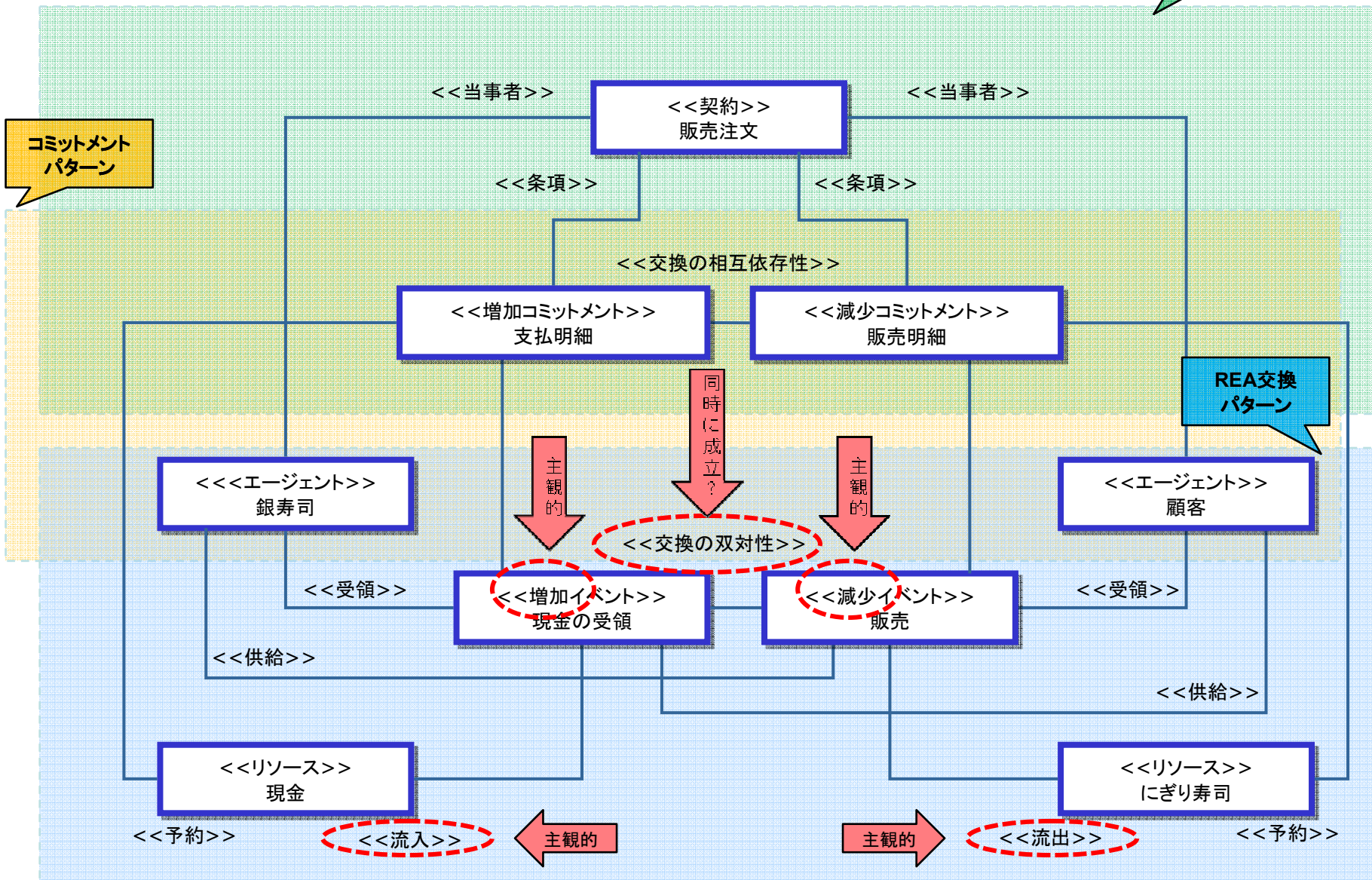


銀
寿
司



REAモデルの例(1)

契約
パターン



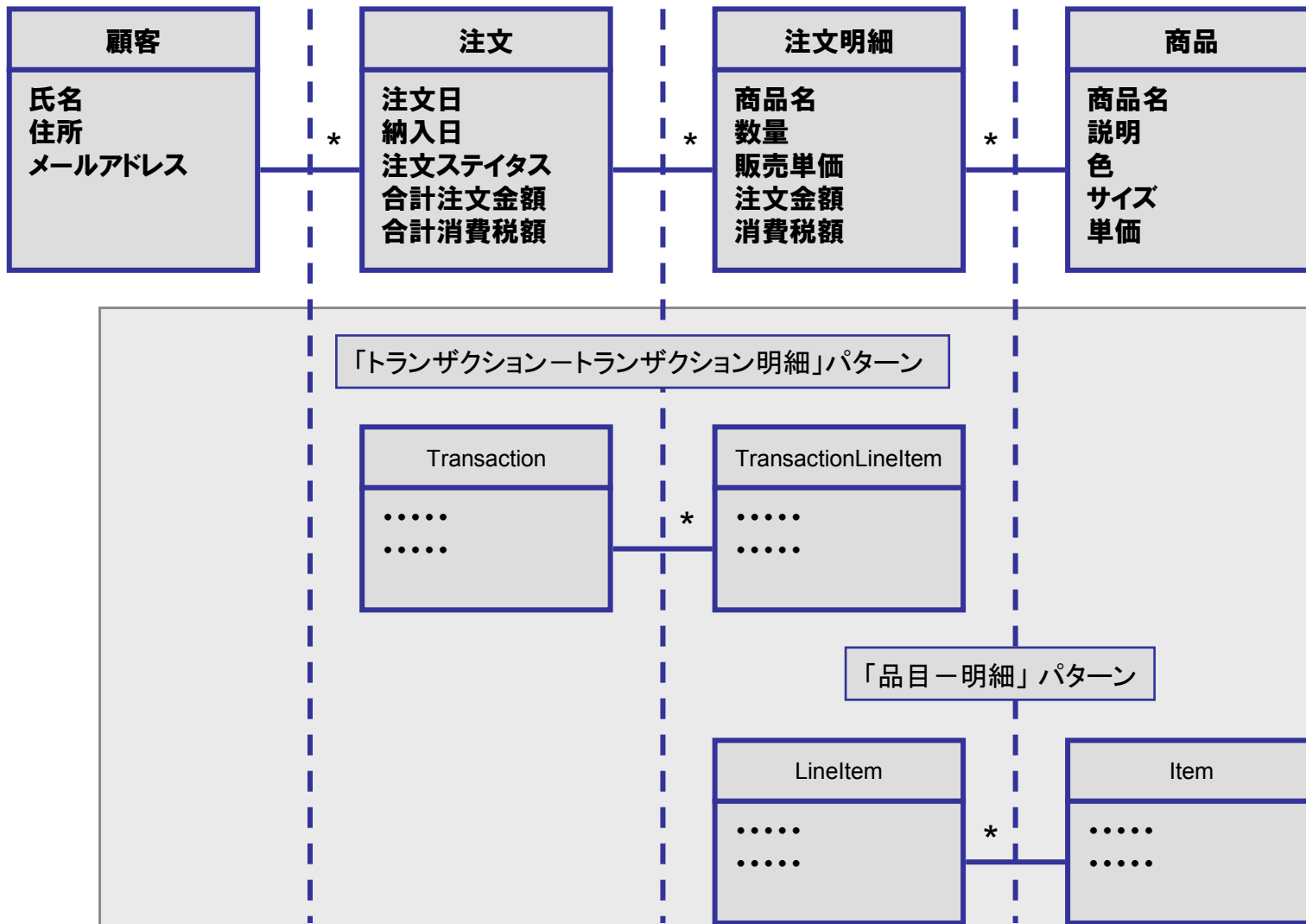
REAパターンを利用するメリット

1. モデルがわかりやすくなる
2. モデルが作りやすくなる
3. モデルを立体的に評価できるようになる

カラーUMLとの対応

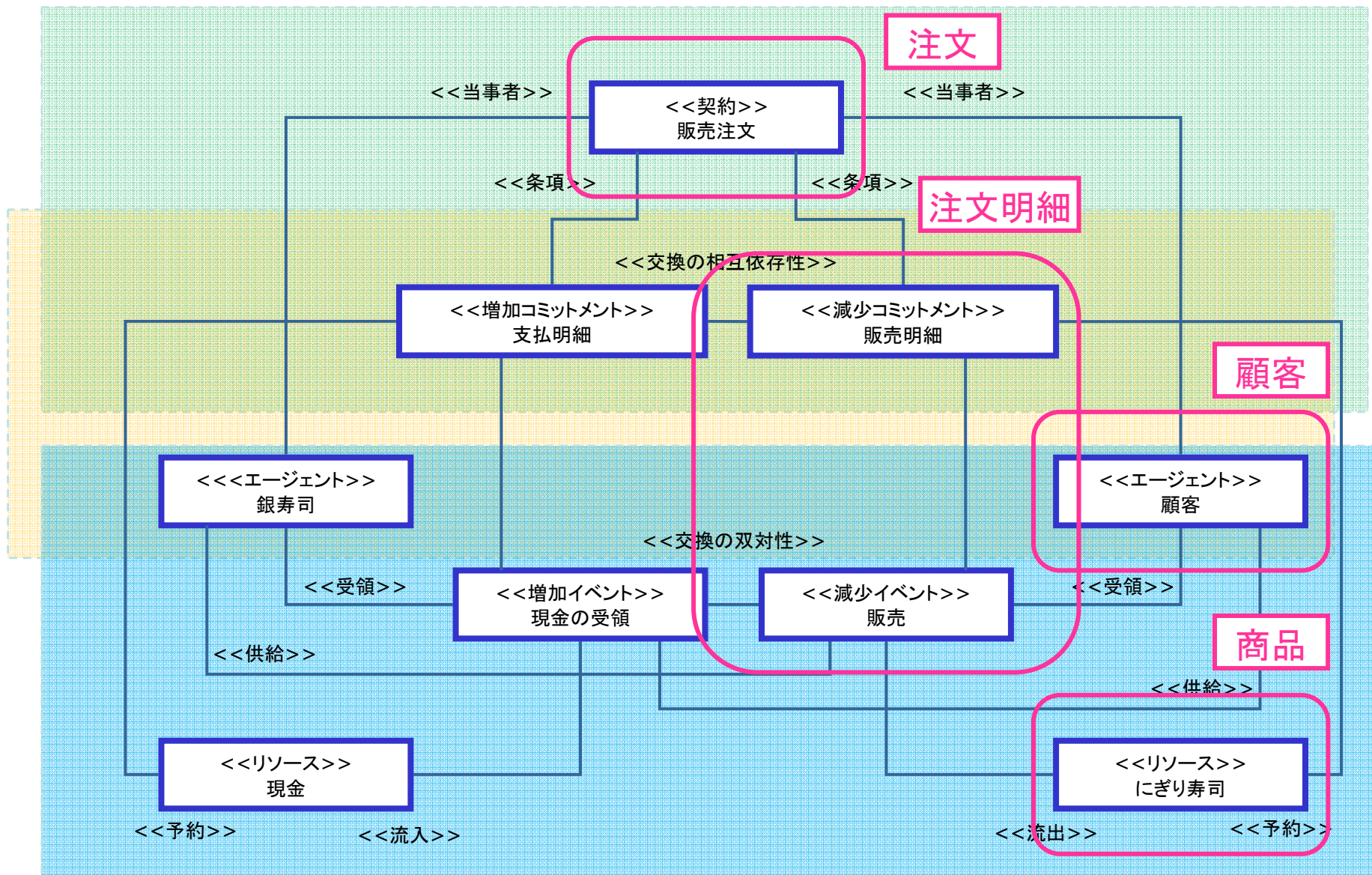
カラーUML		瞬間・時間間隔	役割	パーティー・場所・もの	説明
		ピンク	黄色	緑	青
REA					
構造パターン	業務レベル (REA)	リソース		○	
		イベント	○		
		エージェント		○	
	方針レベル	タイプ	Javaエンタープライズ・コンポーネント(依田他監訳、ピアソン・エデュケーション、2000年)でクラスに色づけをするアイデア(カラーUML)が紹介されている。 http://www.synergy-res.co.jp/publication/book04.html		○
振る舞いパターン		説明			?
		注釈			?

受注出荷システムのモデル

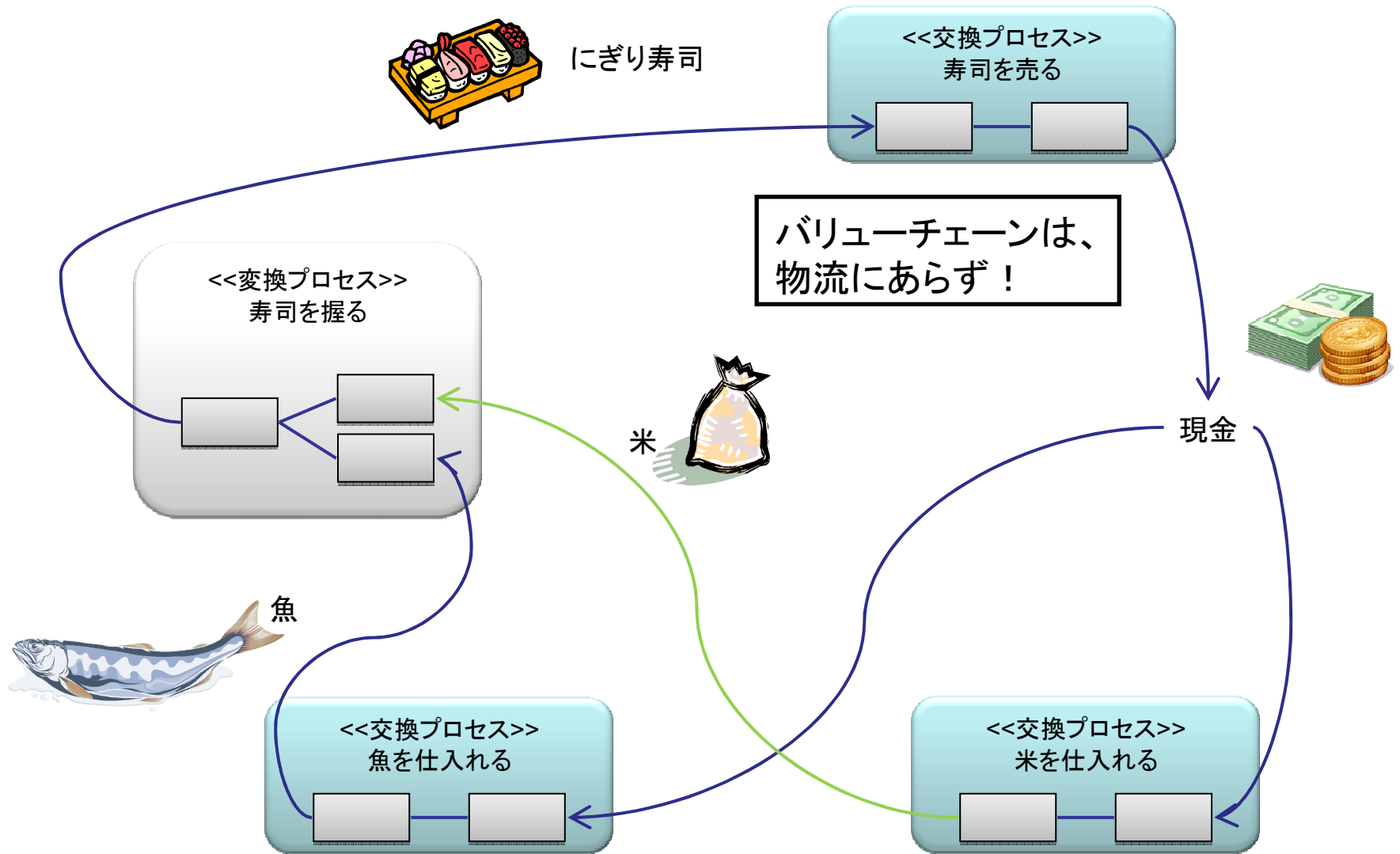


ピーター・コード「ビジネスオブジェクトモデリング」に登場するパターンとの対応

REAモデルの例(2)



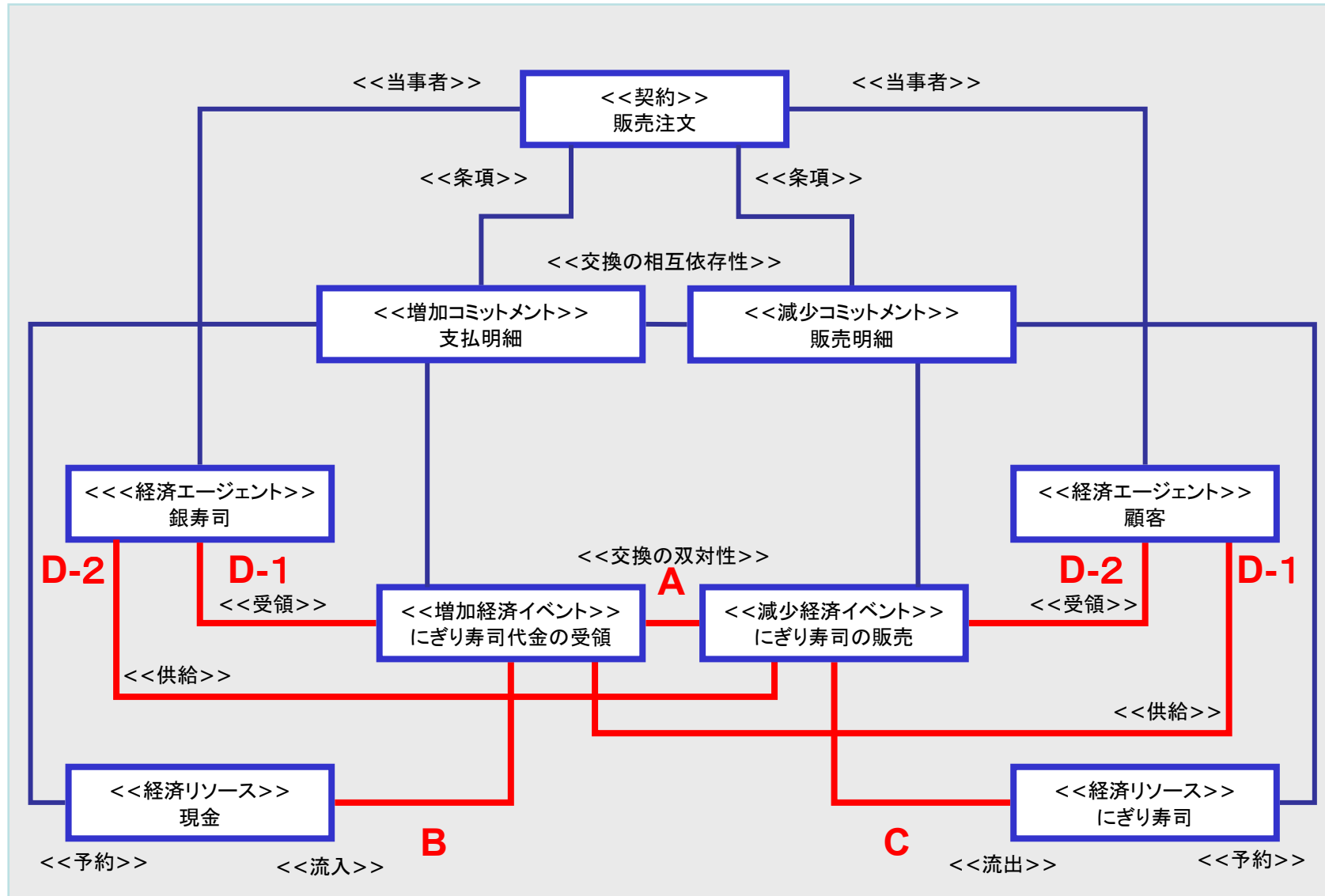
バリューチェーン



REA交換プロセスパターンのためのドメインルール

1. あらゆる増加経済イベントは、交換の双対性にしたがって、1つの減少経済イベントに関連しなければならない。逆もまた同様である。
2. あらゆる増加経済イベントは、流入流出関係により、1つの経済リソースに関連しなければならない。
3. あらゆる減少経済イベントは、流入流出関係により、1つの経済リソースに関連しなければならない。
4. あらゆる経済イベントは、供給関係によって1つの経済エージェントに、また受領関係によって1つの経済エージェントに関連しなければならない。実行時には、これら二つのエージェントが異なる経済的利益を持つ異なるエンティティを表さなければならない。

モデルとドメインルール



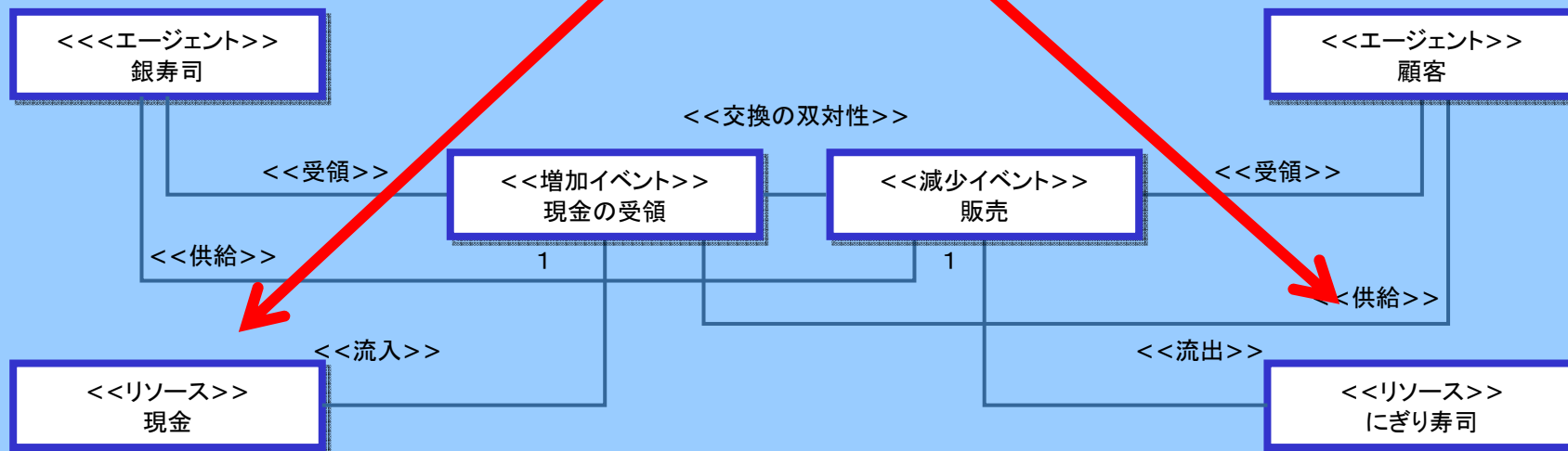
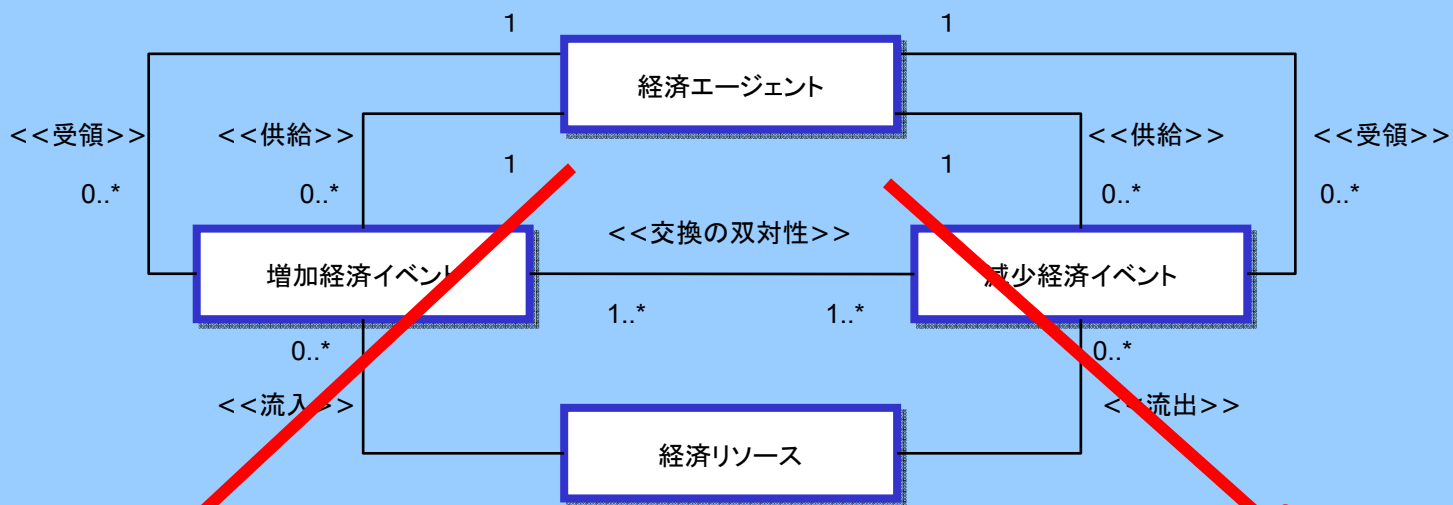
構造パターンを利用したモデリングの勘所

- A) モデリングの目的を意識する
- B) 構造にするか制約にするか
- C) リソースかリソースタイプか
- D) 賃貸と融資
- E) 価値の流れ(廃棄処理)
- F) サービス化
- G) 大規模なモデルへの対応

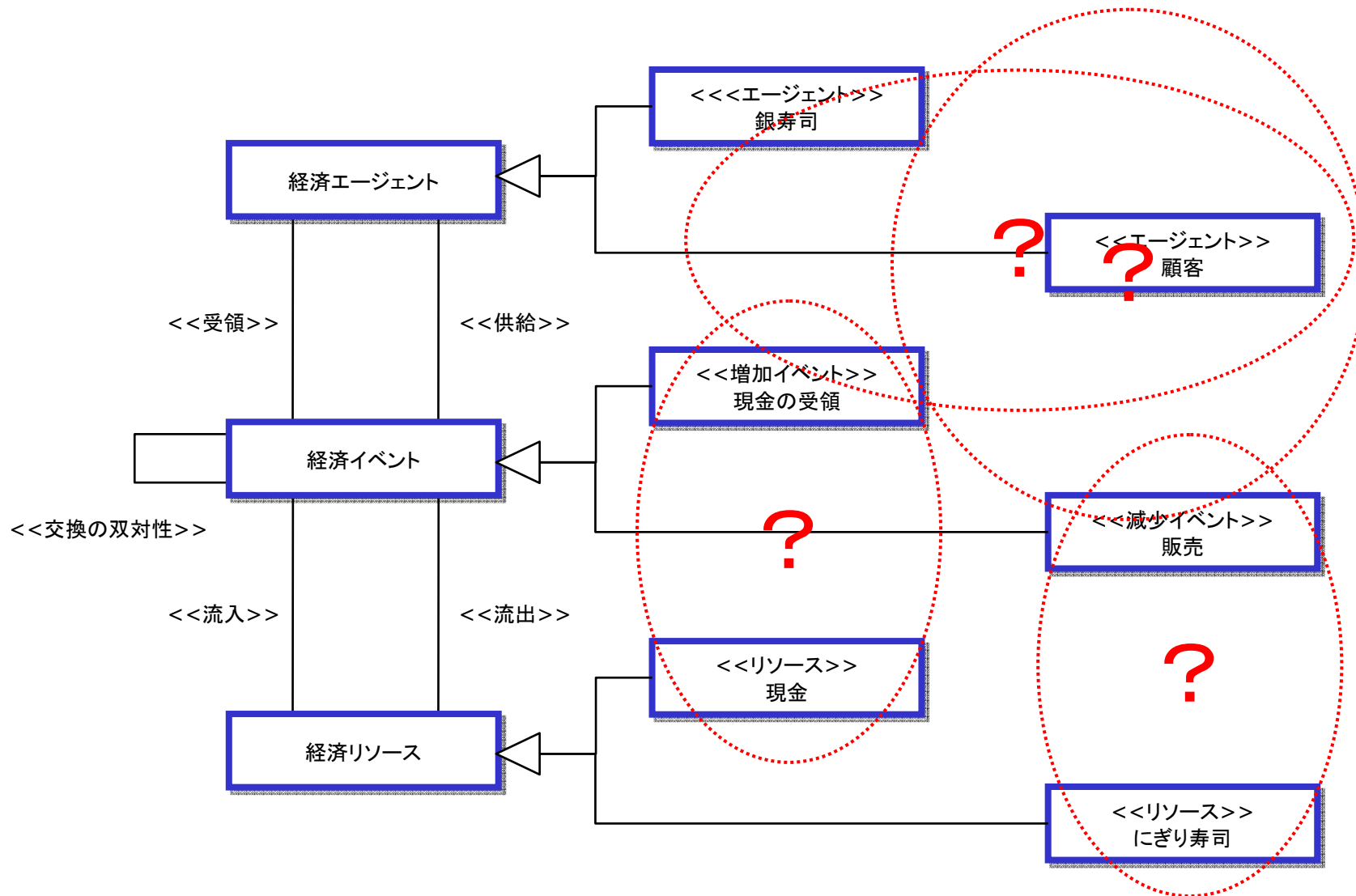
(A) モデリングの目的を意識する

- REAモデリングの目的は、…
 - 何が起きたのか？
 - なぜそれが起きたのか？
 - 何が将来起きるのか？
 - なぜそれが起きるのか？を、知ること
- 上の4つを知ることが自分にとって必要不可欠であると感じ、そのためにはどのような情報が提供されればよいのかを頭に描いてモデリングすれば良い
- 経営者の気持ちになれば理解しやすい…
 - 「いったいこの金は何に使ったんだ！」
 - 要するにERPS(Enterprise Resource Planning System)を作る心持ち？

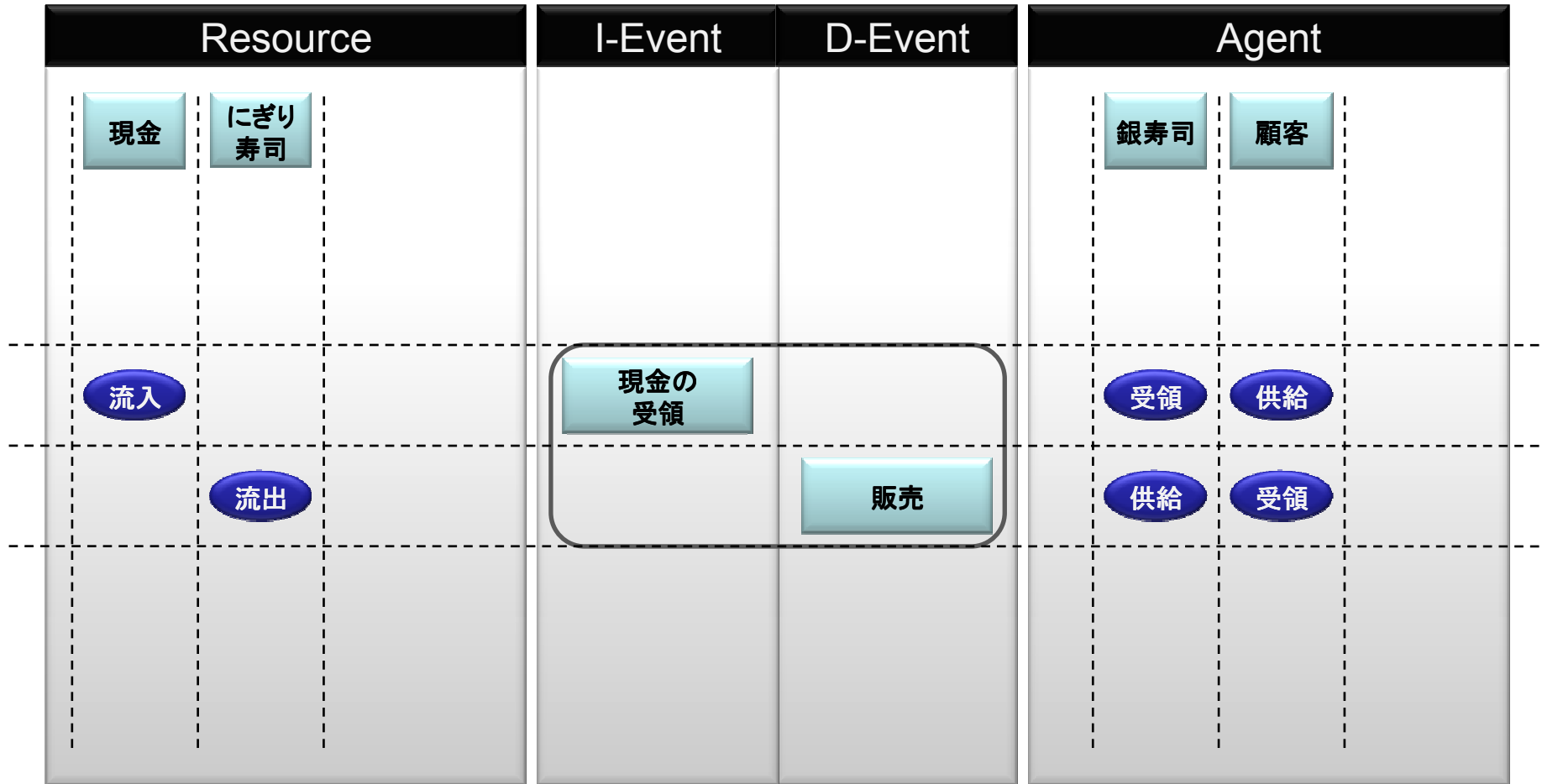
(B) 構造にするか制約にするか



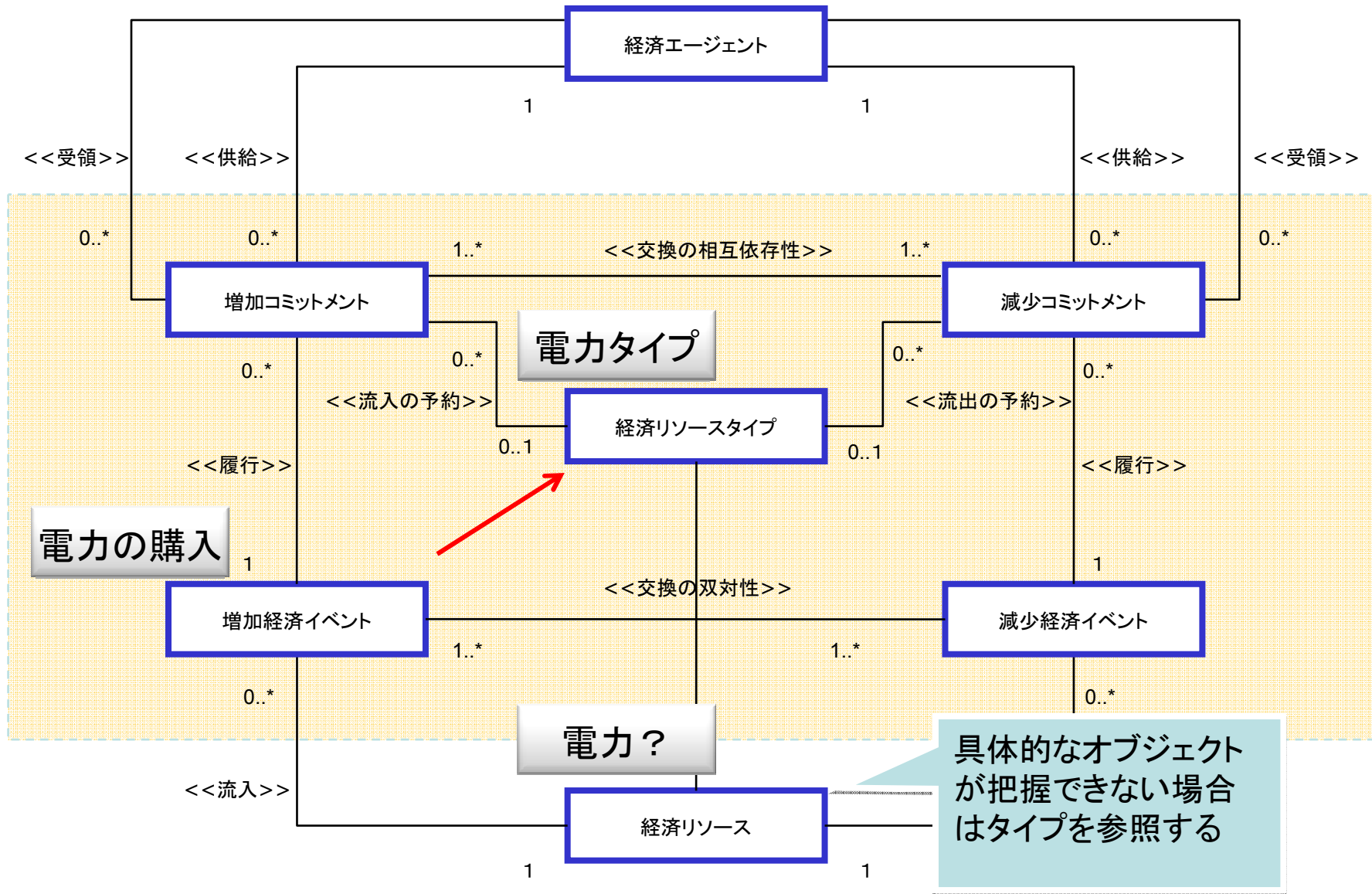
REAをフレームワークとしてとらえると



制約のマトリクス表現

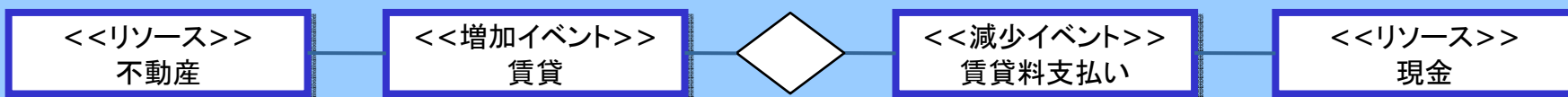


(C) リソースかリソースタイプか



(D) 賃貸と融資

<<交換の双対性>>



借りた不動産
は返す

理由:

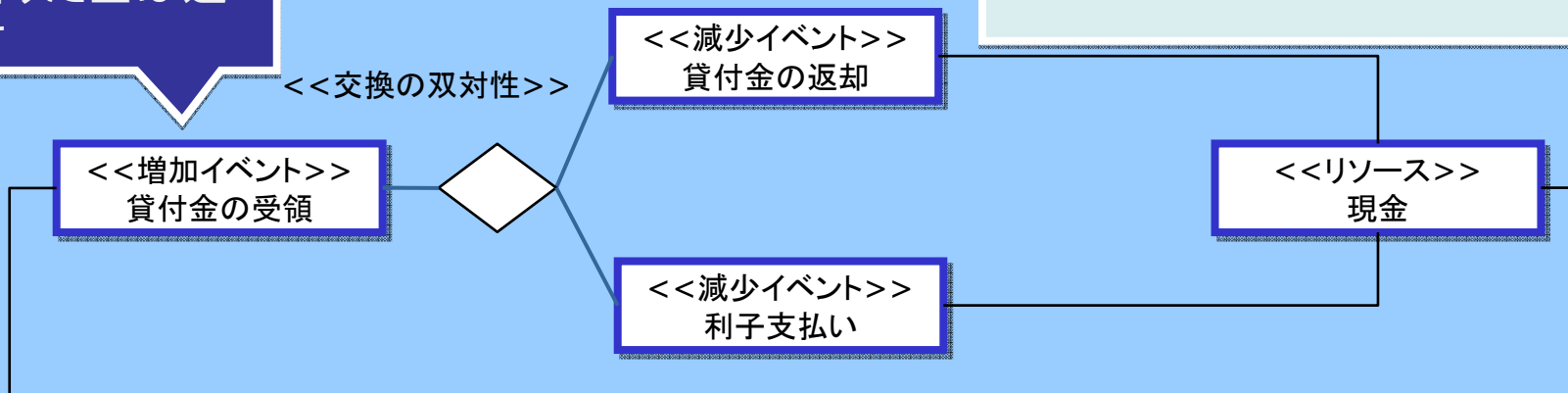
賃貸では同じ不動産を返さなければならないが、
融資は、小切手で借りて現金で返しても良い。

現金と不動産の区別はつくが、借りた金を現金の
区別はつかない。

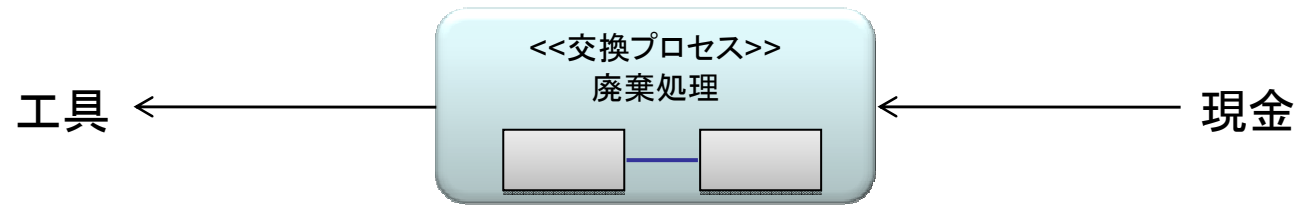
不動産は分割して返せないが、金は返せる。

借りた金は返
す

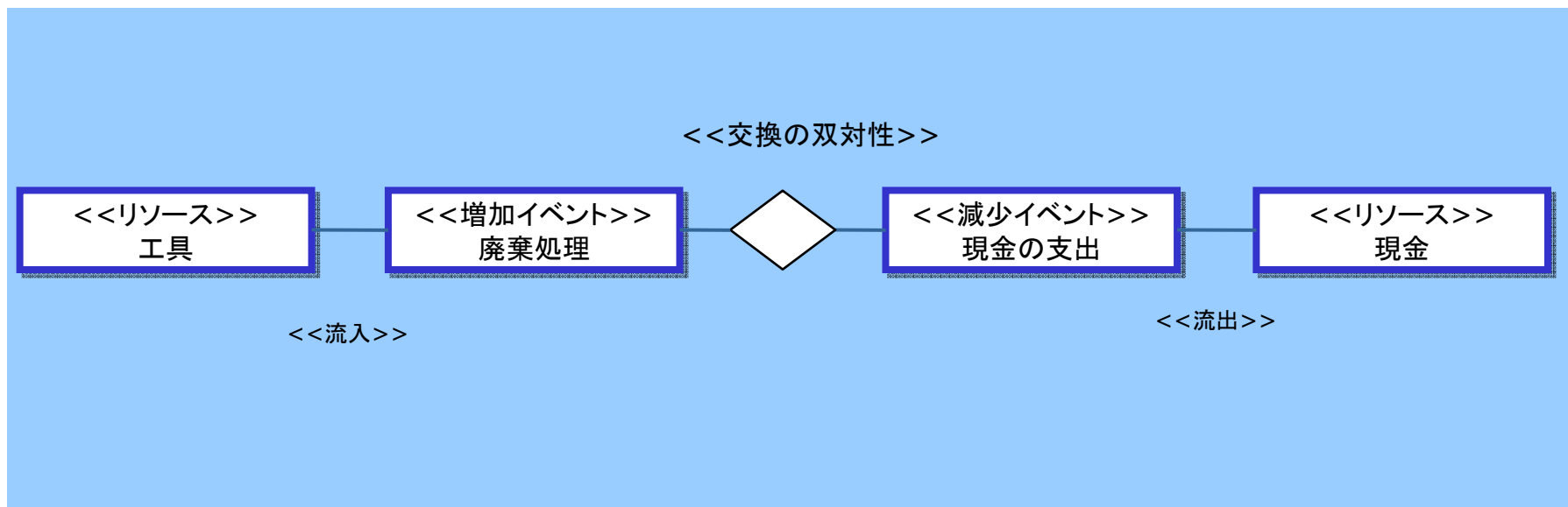
<<交換の双対性>>



(E) 価値の流れ(廃棄処理)

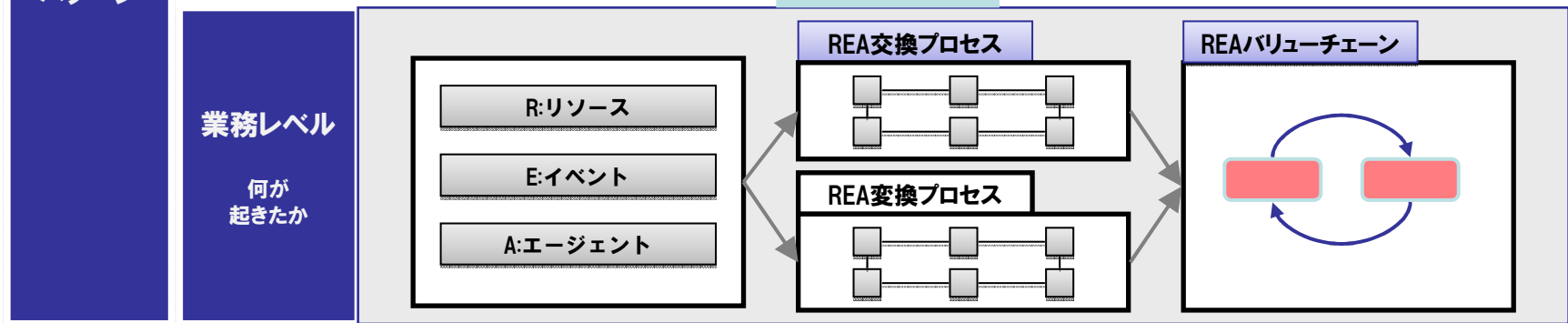
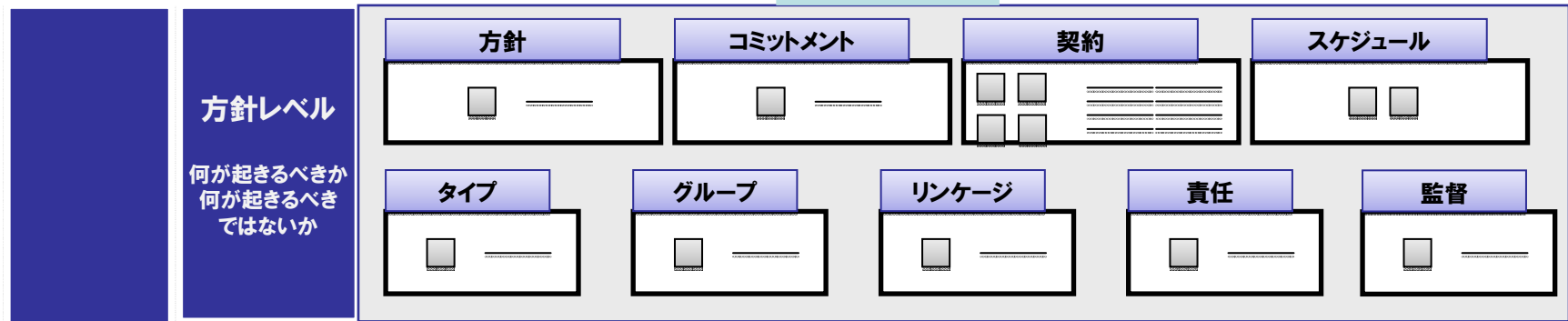
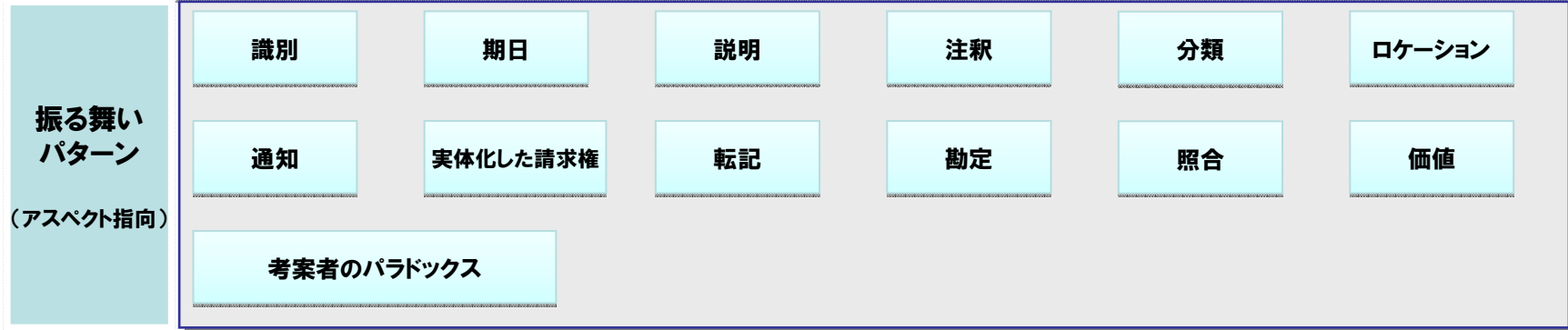


バリューチェーンは物流にあらず

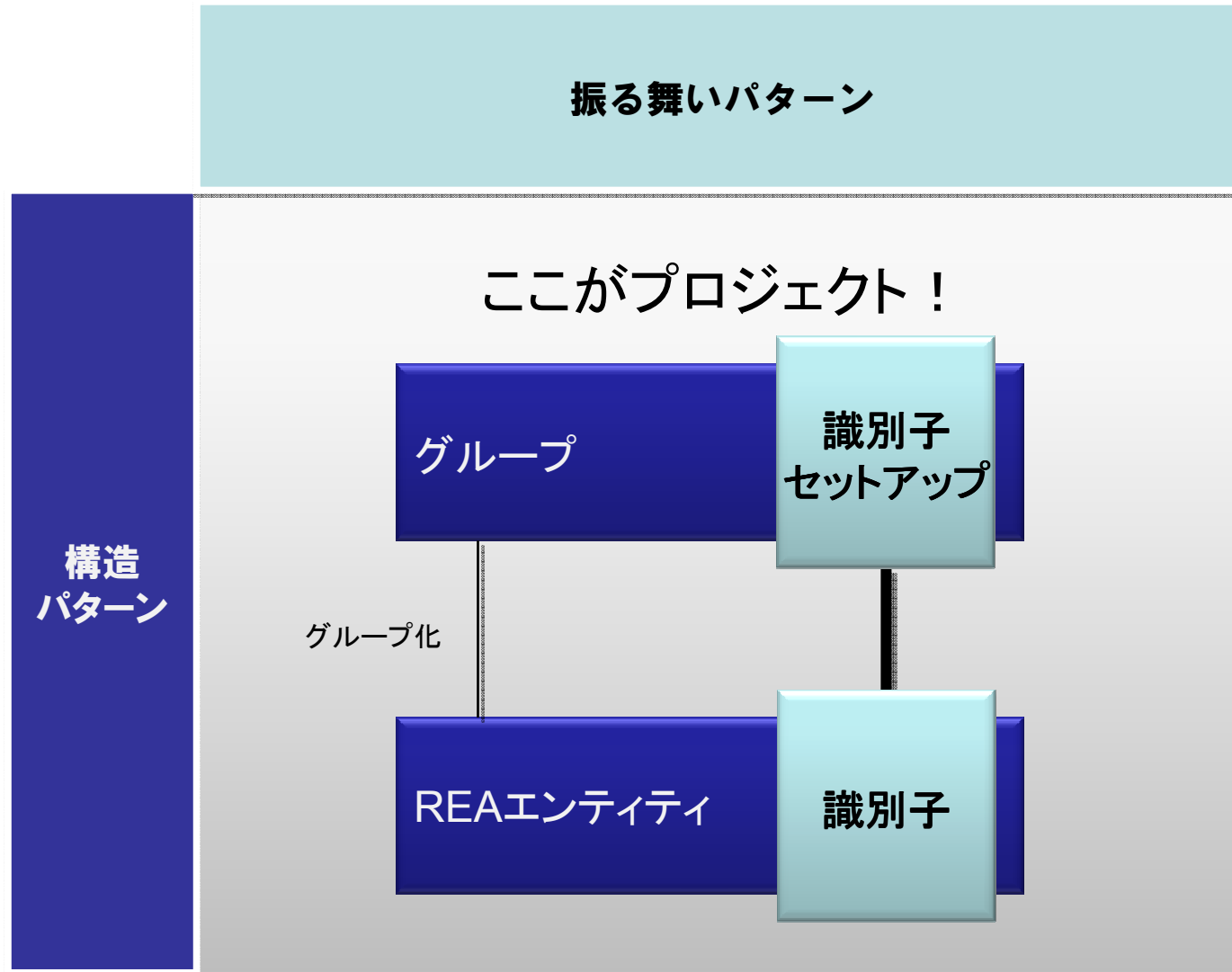


質問コーナー

構造パターンと振る舞いパターン



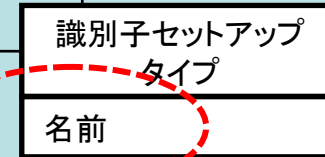
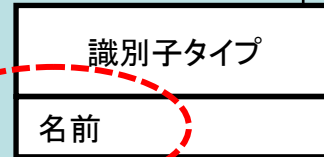
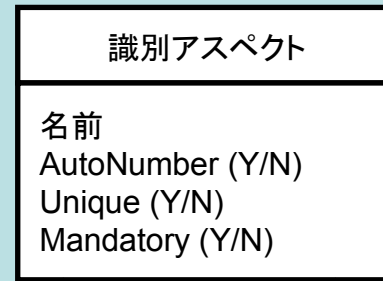
構造パターンと振る舞いパターンは直交している（識別パターンの場合）



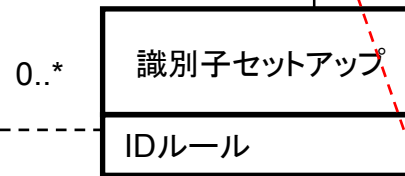
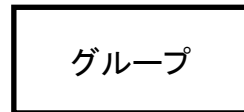
識別アスペクト

アプリケーションモデルの上位にクラスが定義されていることが重要。
つまり、モデルに対して横断的だと言うこと。

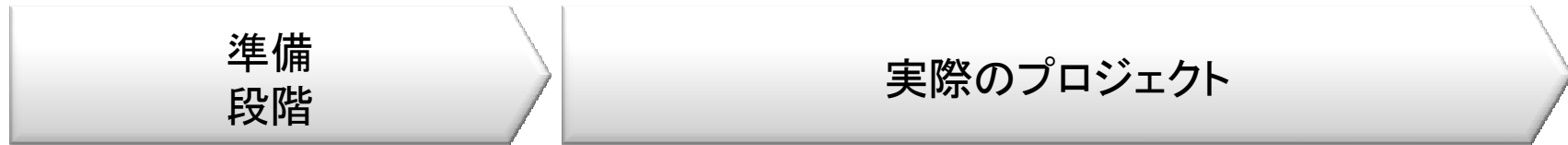
アスペクトタイプ



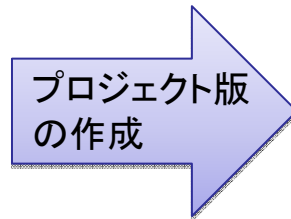
アプリケーションモデル



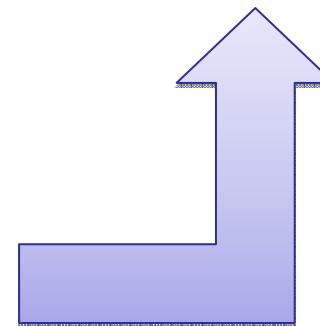
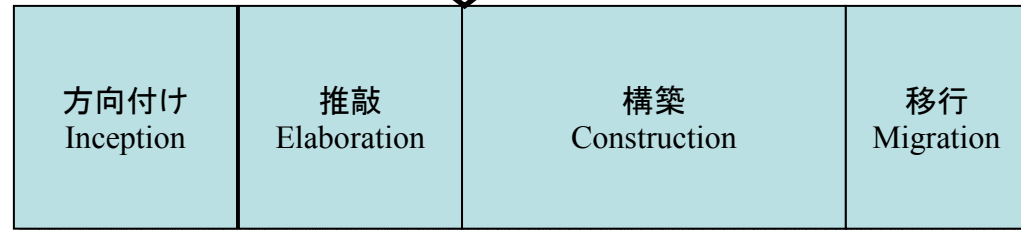
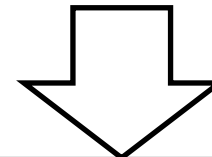
REAとプロジェクト



研究会
などで
自社版
ビジネスパターン
を作ってみる



いきなりこのあたりから
始まっていないか。
それは儲からない！

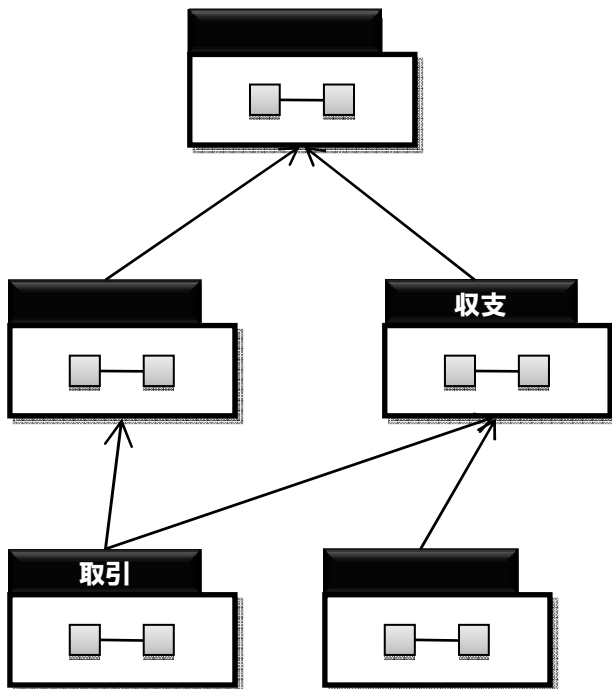


本格展開による
量産効果をねらう

ビジネスパターンへの私の取り組みの歴史

10年ほど前に、CBOP(ビジネスオブジェクト推進協議会)のBFOP(Business Function Object Patterns)開発に参加。

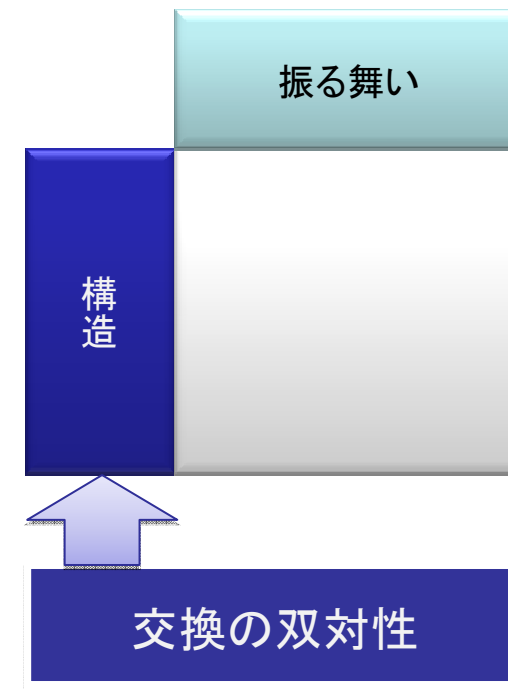
UMLのパラメトリック・コラボレーションによる1次元的な組み立て構造



REAの場合:

構造パターンと、振る舞いによる2次元的組み立て構造

基盤部分を交換の双対性が支える



開発プロセスにもたらす効果

- 従来からの要求分析や上流設計の手法は業務プロセス主導型である。
- そのため、ユーザーの関心はキープできるものの、システム設計に有用な情報を提供できない。
- REAは、あらかじめ、構造パターンと、振る舞いパターンが用意され、構造モデルはバリューチェーンモデルと直結している。従って、バリューチェーンモデルが描ければ、構造モデルを作成し、振る舞いパターンの合成などにより、一定のシステム開発作業に直接進むことができる。
- バリューチェーンモデルは、仮想化、サイバー化、Web2.0化する社会におけるビジネス構造を骨太に描き出すツールであり、ユーザーの関心も得やすい。
(物流モデルとは違う！)
- バリューチェーンモデルは、リソースとプロセス(業務)の関係を表している。従って、バリューチェーンモデルだけで、企業のリソース計画管理に関する要件を相当程度表し得ると言えるのではないか。
- 以上の考察から、REAはERP(Enterprise Resource Planning)的システムの要件定義と設計開発手法として大きな可能性を持っていると思われる。

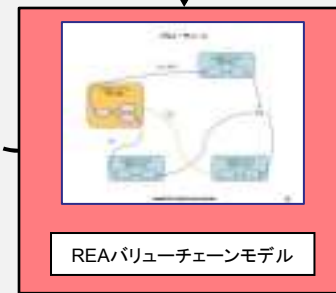
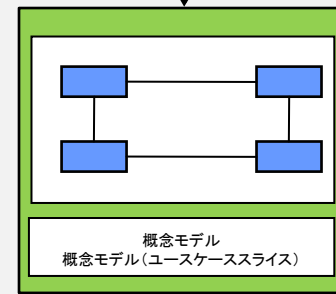
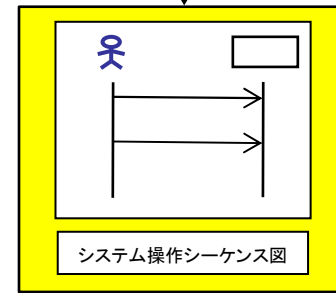
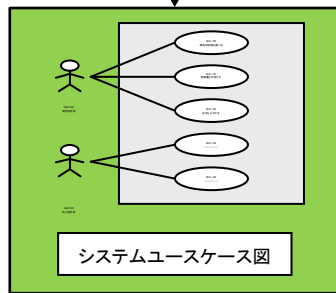
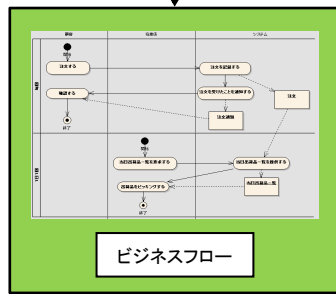
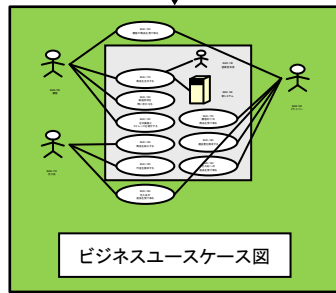
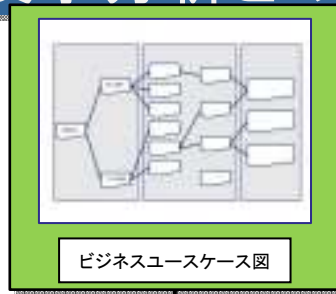
要求開発、要求分析との関係



要求開発
(Openthologyなど)手
法が提唱するツール

要求開発のような
トップダウンア
プローチも欠かすこ
とはできないだろう。

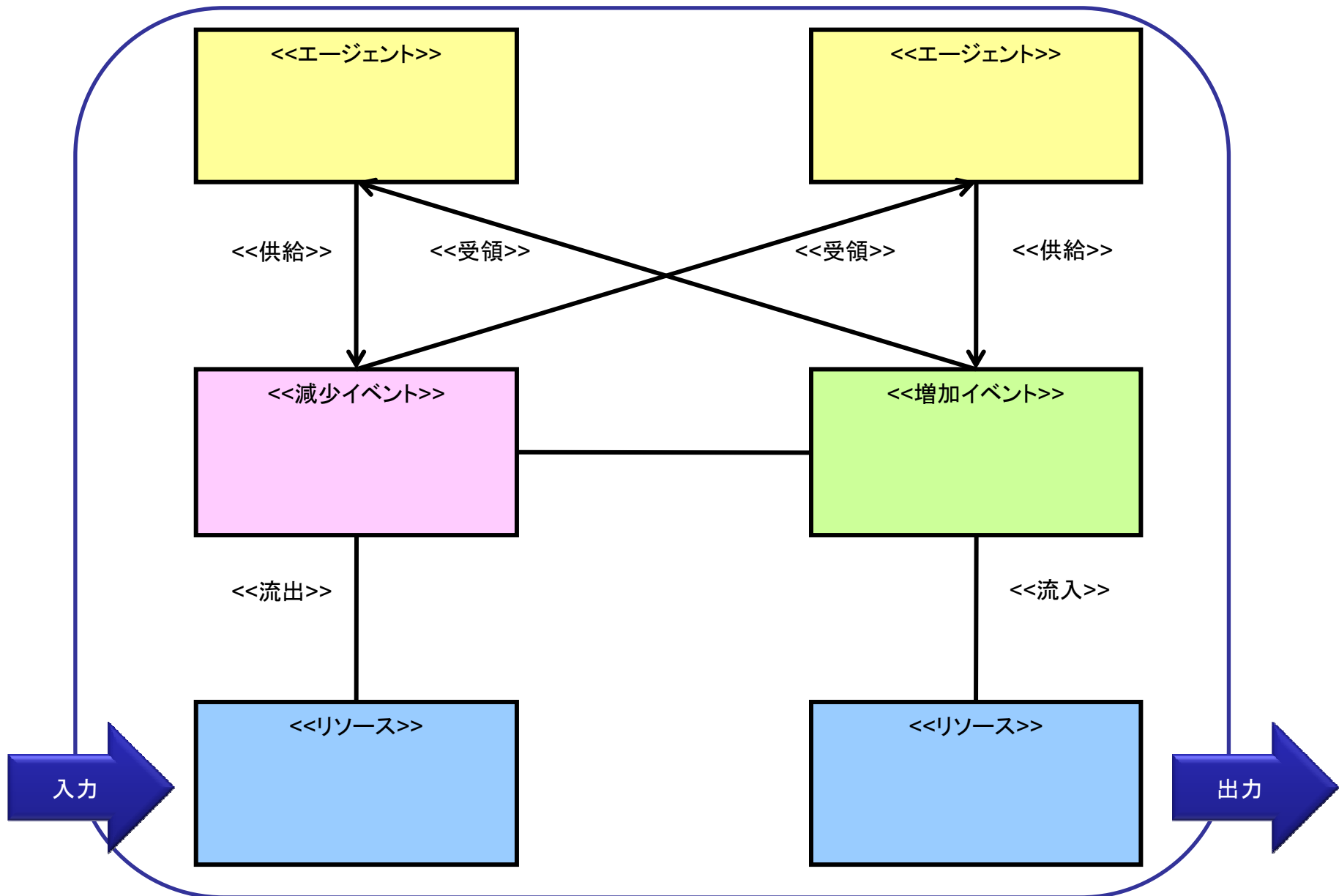
REAのようなビジ
ネスの普遍性に基
づくアプローチとど
のよう合流させれ
ば良いか。



UML的な進め方
(参考文献)
実践UML(ピアソン・エ
デュケーション)
2007年11月第3版
刊行
初版解説ページ
<http://www.synergy-res.co.jp/publication/book02.html>

REAが構造パター
ンとして提唱する部
分

モデリングツール(REA交換プロセスカード)



REAモデリング演習

- 最近行った取引をモデル化してください
 - 個人でひとつ
 - ビジネスでひとつ

 - プライバシーや機密にはふれない範囲で..

参考情報

- <http://reatechnology.com>
- <http://www.synergy-res.co.jp>