

**DX推進の切り札！**

**業務の関係をデザインするための  
リレーショナルプロセスモデリング**

羽生 章洋

# 自己紹介

## 羽生 章洋 (はぶ あきひろ)

1968年生まれ 大阪府出身



1989年、桃山学院大学社会学部社会学科を中退後、  
2つのソフトウェア会社にてさまざまな業種・業態向けシステム開発に携わる。

その後、アーサーアンダーセン・ビジネスコンサルティング（当時）に所属。  
コンサルタントとして企業改革の現場に従事。

さらにその後、 트레이ダーズ証券株式会社（当時）の新規創業にIT部門ディレクターとして、  
さらにマネースクウェアジャパン株式会社（当時）の新規創業にIT担当取締役として参画、  
両社にて情報システム部門の構築と、日本初のリッチクライアント+OSSによる  
オンライントレーディングシステムの企画構想から要件定義/設計/開発/運用保守までを行う。

2006年から2011年まで、国立大学法人琉球大学の非常勤講師。  
現在はエークリッパー・インク代表を務めながら、企業向けにIT化の推進や業務設計の支援などを行っている。  
業務フロー作成ツール「マジカ」やシステム要件定義図法「IFDAM」の作者である。  
著書多数。



# DX、進んでますか？

- 順調なら、このあとの話は聞かなくてもいいですw
- 上手く進まない原因

細かいことは色々と挙げられますが…

**「業務プロセスをデザインできない！」**

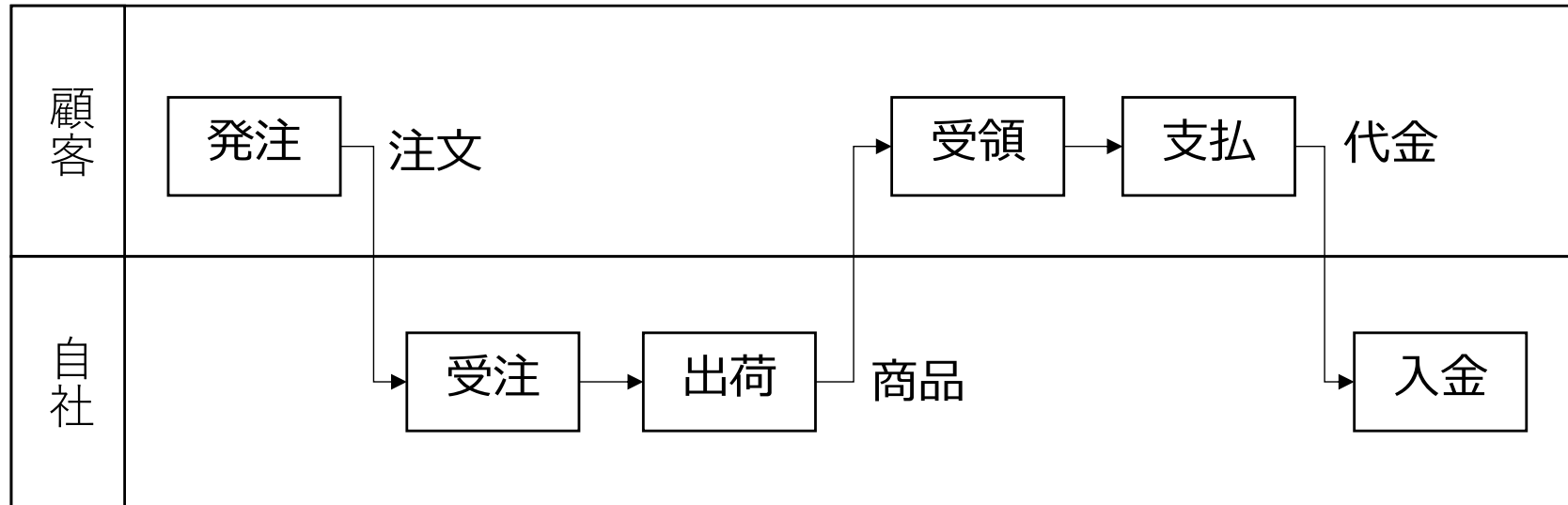
これに尽きる！

- じゃあどうして業務プロセスをデザインできないのか？

これも理由は色々とあるんだけど…

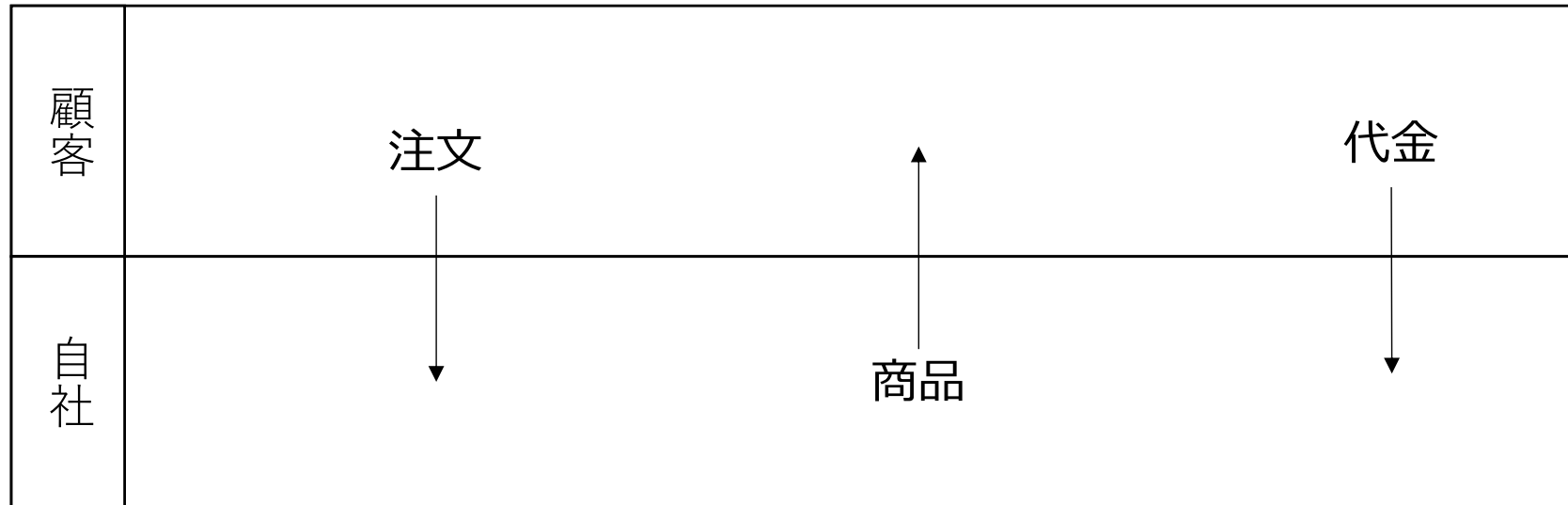
そもそも **「見るべきポイントがズレている」**！

# よくある業務フロー



みんな個別の仕事（アクティビティ）に注目しちゃうんだけど…

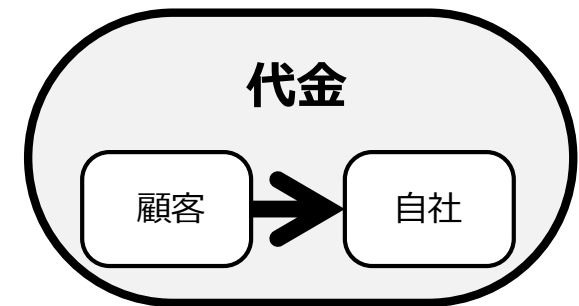
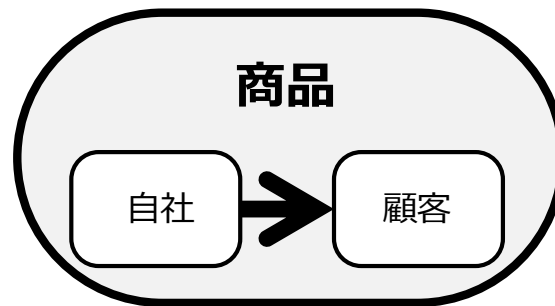
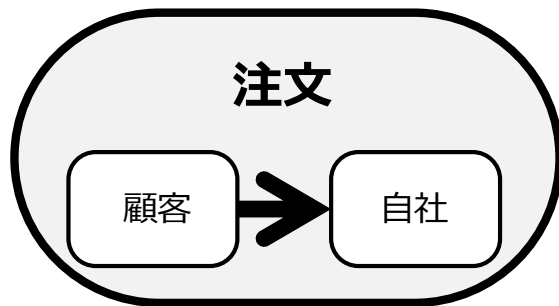
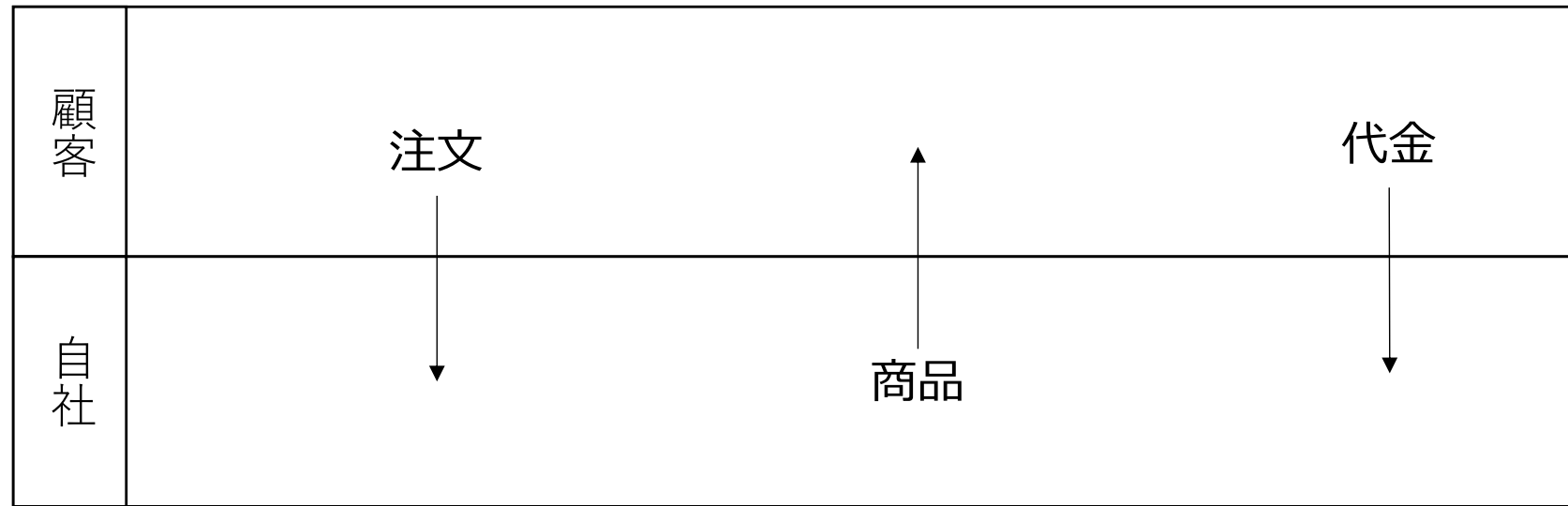
# 本当に重要なのはこれじゃね？



これがちゃんとできないとマズいのでは？  
怒られたり、クレームになったり…

→受け渡しこそが重要なのでは？

# 受け渡しに注目してモデル化してみる



このひとつずつの受け渡しを業務における「関係」と呼ぶ

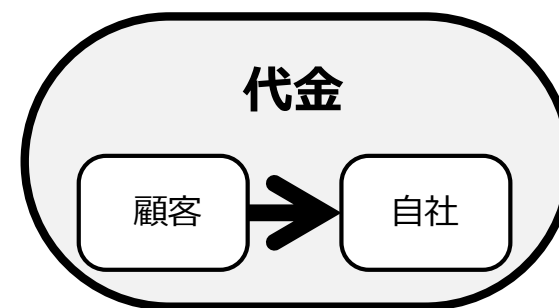
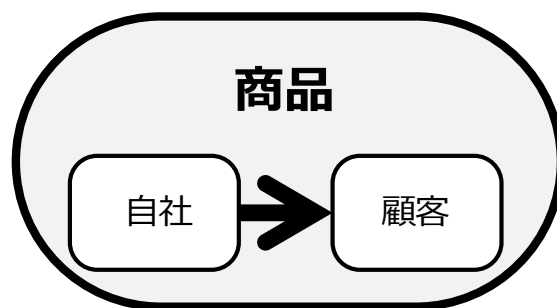
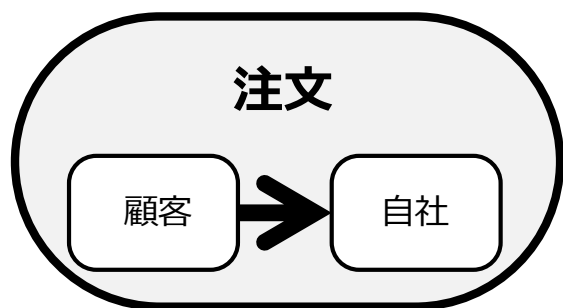
# 業務における関係

## 関係とは

- ・ 受け渡しであり
- ・ やり取りであり
- ・ 取引であり
- ・ インターフェイスであり
- ・ 境界線であり
- ・ メッセージングであり...



依頼に対する応答であり  
需要に対する供給であり  
期待に対する**責任**である



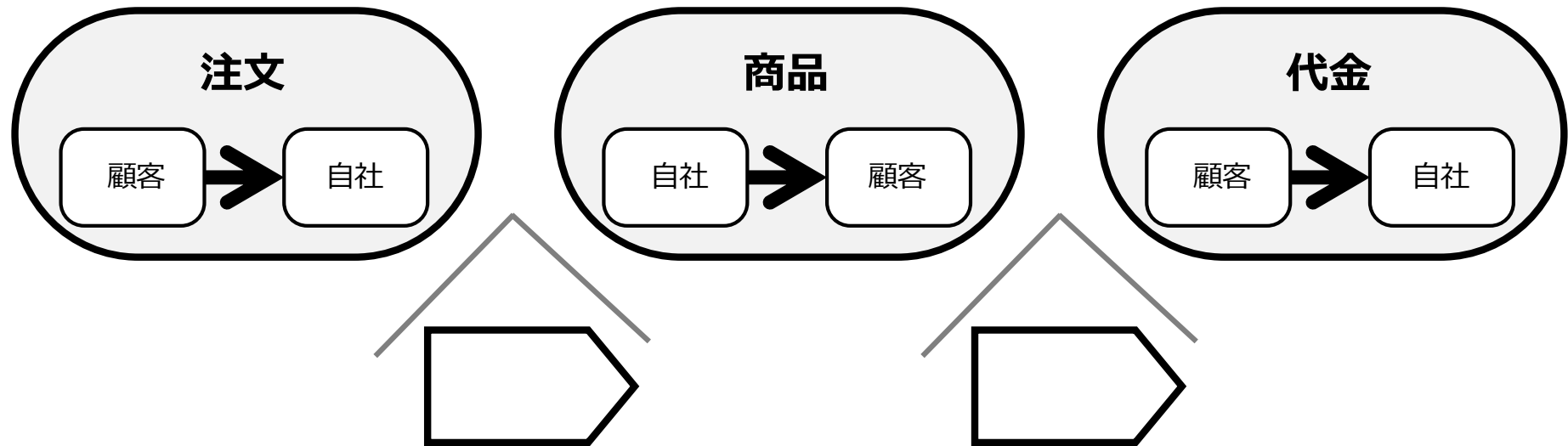
業務における関係には責任が伴う  
…分業における責任分界点

# リレーショナルプロセスという考え方

業務における関係には責任が伴う

→責任を果たすためには

関係を成立させるために個別の仕事を行う必要が生じる



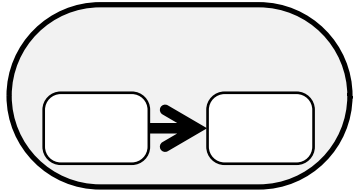
関係（リレーション）に注目してその関係を成立させるために  
どんな仕事が必要なのかという業務プロセスの構造を考える  
…リレーショナルプロセスモデリング



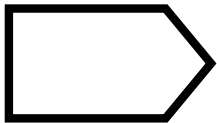
# デジマシによるモデリング実践のデモ

# そしてWagbyへ！

デジマジによる  
プロセスモデリングから...



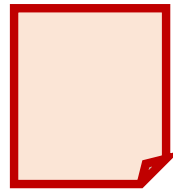
この関係を成立させるために  
こんな業務が必要だ



この業務を実現するために  
こんなシステムが必要だ



**このときに必要な  
データ項目と処理はこれだ！**



データ  
モデリング  
して...

Wagbyデザイナーへ  
そして実際に動くシステムへ！

No.	項目名	型	検索	一覧	モデル項目説明
1	ID	整数			
2	項目1	文字列			
3	項目2	文字列			
4	項目3	文字列			
5	項目4	文字列			
6	項目5	文字列			

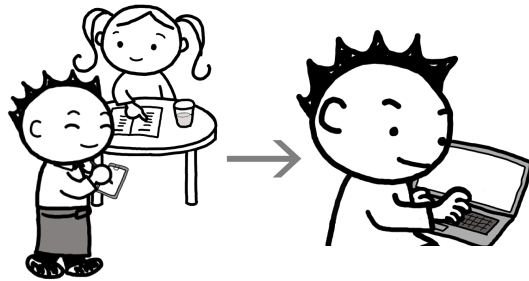


**やりたいプロセスを迅速に実現できる  
ローコード/ノーコード開発の価値が活きる！**

# DXで重要なのは"新しい"プロセスをデザインすること！

例えばレストランの注文登録というシステムを考えるとして…

ホール係が書いた  
注文伝票を  
事務担当が  
入力する



ホール係が  
接客しながら  
注文データを  
登録する



顧客が  
セルフ注文する



どのケースも  
データモデル（注文データ）も  
システムの機能要件  
（注文データを追加する）も  
同じように見えてしまう

↓ しかし！

**やりたい業務プロセスも  
システムで実現したいこと  
（ユースケース）も全く異なる！**

↓ だから

**まず最初に  
「今とは違う」「新しい」  
プロセスのデザインが必要！**

## まとめ

**DX = SoRからSoEへ  
Record (記録) から  
Engagement (つながり) へ**

**つながり = 関係に着目して  
業務プロセスを  
しっかりデザインしよう！**