

地に足のついたDXを実現するための メッセージ指向の業務デザイン

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
- DX、ぶっちゃけよわからん
- でもせっかく上が「DXやろうぜ！」とか言ってるならチャンスよね
- とはいえ、効率化とか生産性向上とか無駄を無くすとか言っても
 - 具体的に何をどうすりゃいいの？
 - 何がどうなったら、そういう結果を出せたといえるの？
- 地に足のついたDX =

「システムによって、仕事における人間のストレスを撲滅する！」

と定義してみる

- まあぶっちゃけシステムを入れたからストレスが増えた、とかありますがそういうのも何とかしようぜ、と

仕事のストレスって何だろう？

- 作業のストレス

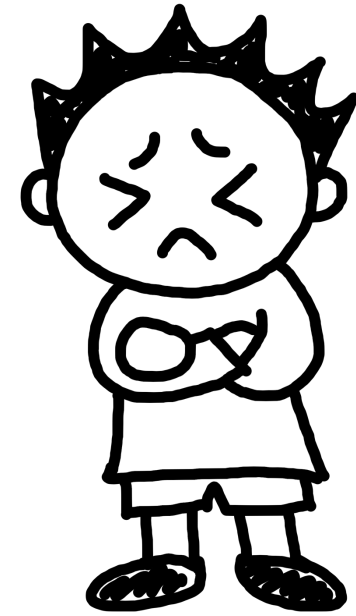
- 難しいとか面倒くさいとか

→そこで自動化！

RPAとか

…去年は

「ノーモアコピペ！ノーモア再入力！」
とか言ったんですよ



- 人間関係のストレス！

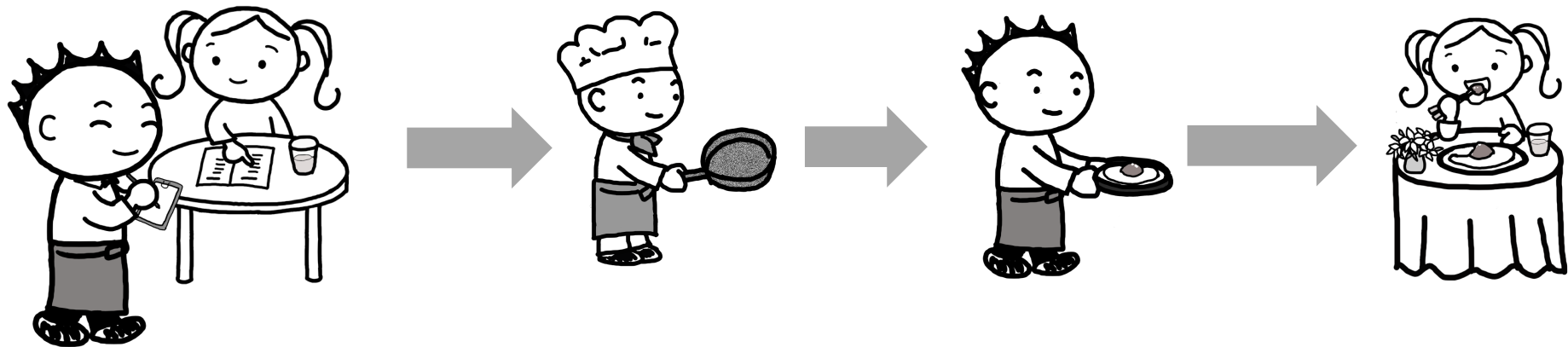
- 言い換えるとコミュニケーション
- コミュニケーション＝情報の伝達／共有

- **「仕事で必要な情報がきちんと伝わってこない！」**

- 気配りや再確認はコスト！

何故、コミュニケーションが必要になるんだろう？

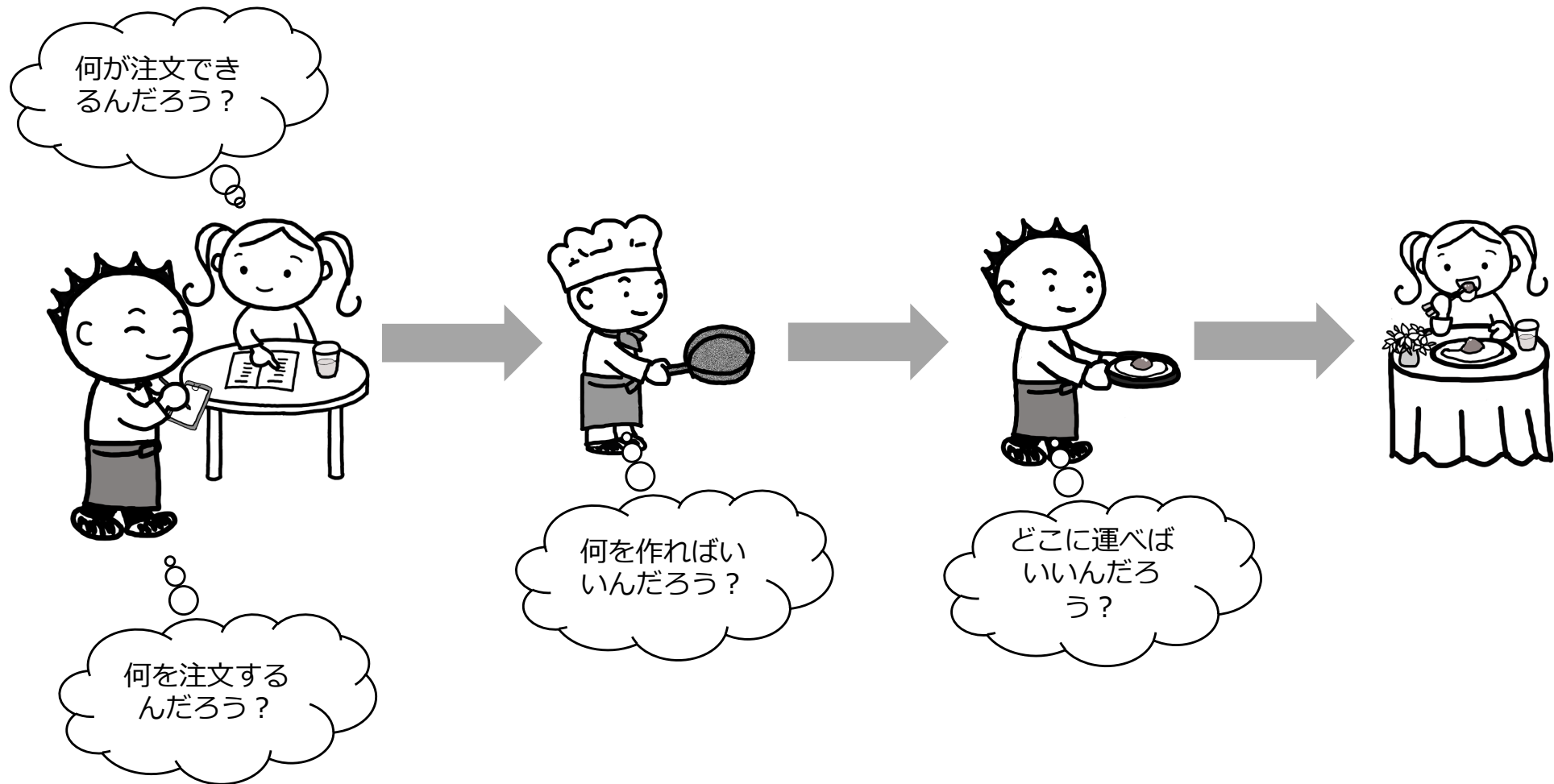
やるべき仕事が増えるほど人手が必要→分業が進む



- 分業するほど、ひとりずつの役割が明確になる→**専門化**の進行
- 自分の仕事に専念することができるので**作業効率がアップ**する

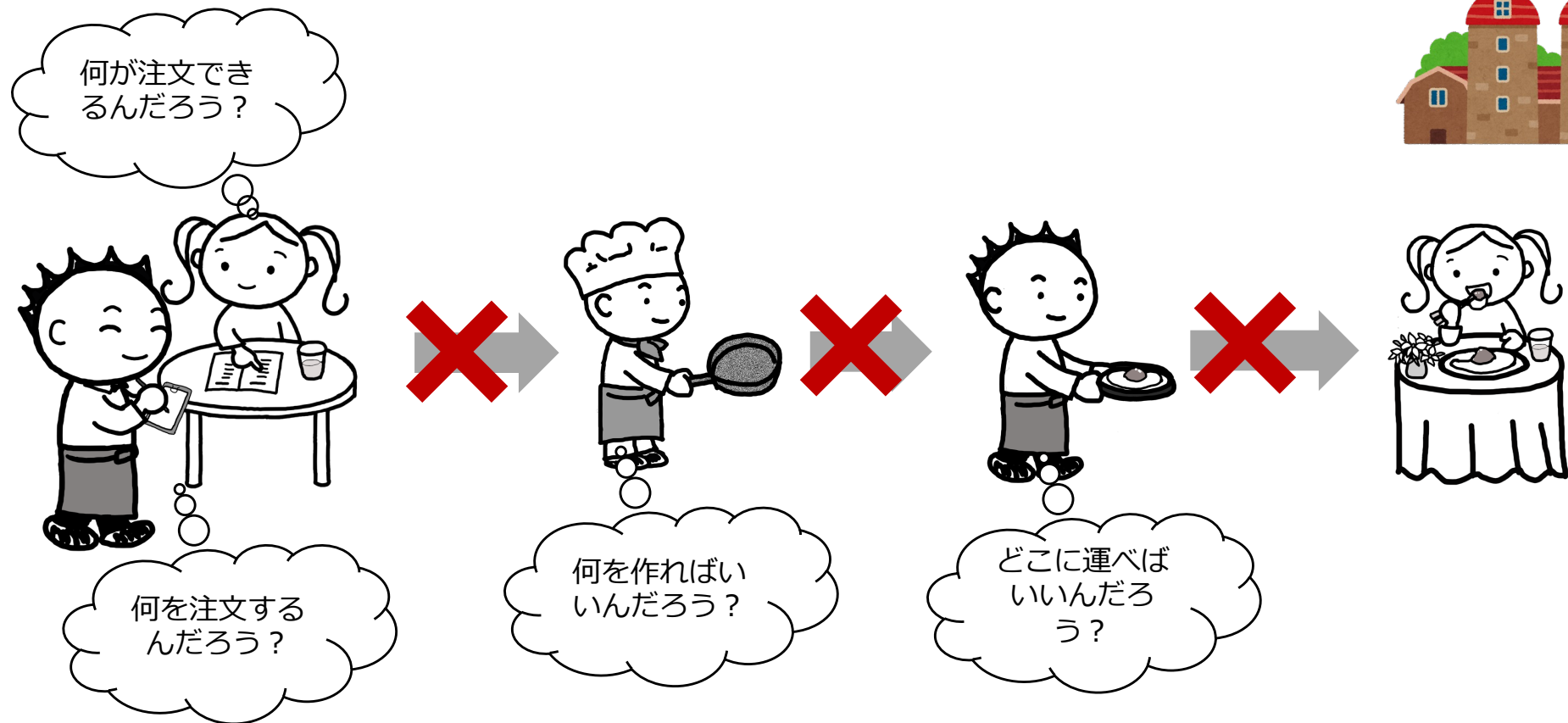
何故、コミュニケーションが必要になるんだろう？

分業が進むほど、それぞれが自分の仕事をするために
「知らないと仕事が出来ないこと」を伝え合う必要が生まれる



もしちゃんと情報が伝わらなかったら？

分業するほど情報伝達（コミュニケーション）の必要性が高まるが、
情報は物理制約を伴わないため扱いが雑になりやすいせいで
情報伝達は面倒くさいので疎かになりやすい→情報の断絶！サイロ化／縦割り



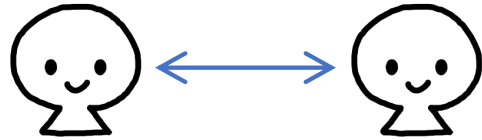
連携プレーが途絶えてしまい、
個々の仕事が優れていても意味がなくなる

分業におけるコミュニケーションパスという問題

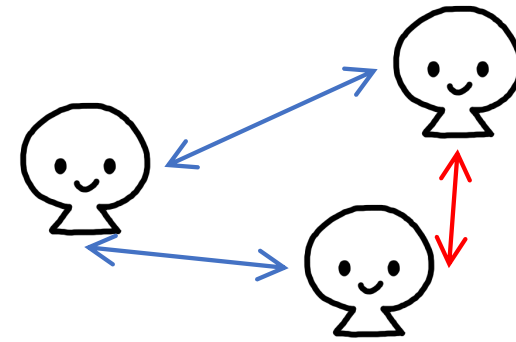
全体のコミュニケーションパス数 = $n(n-1)/2$

一方で自分がカバー出来るパス数 = $(n-1)$

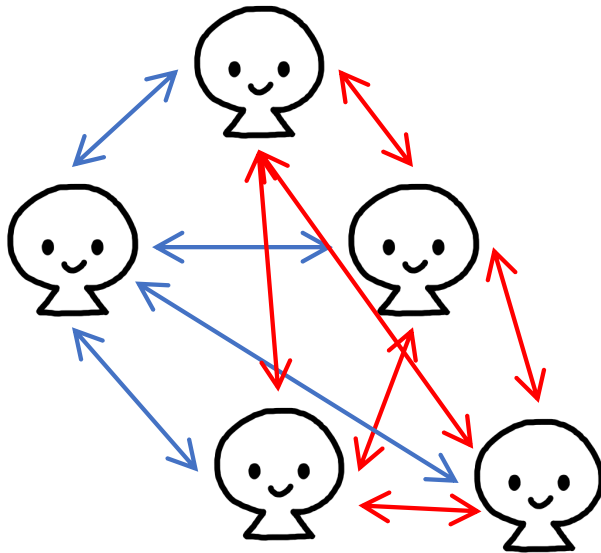
※ nは人数



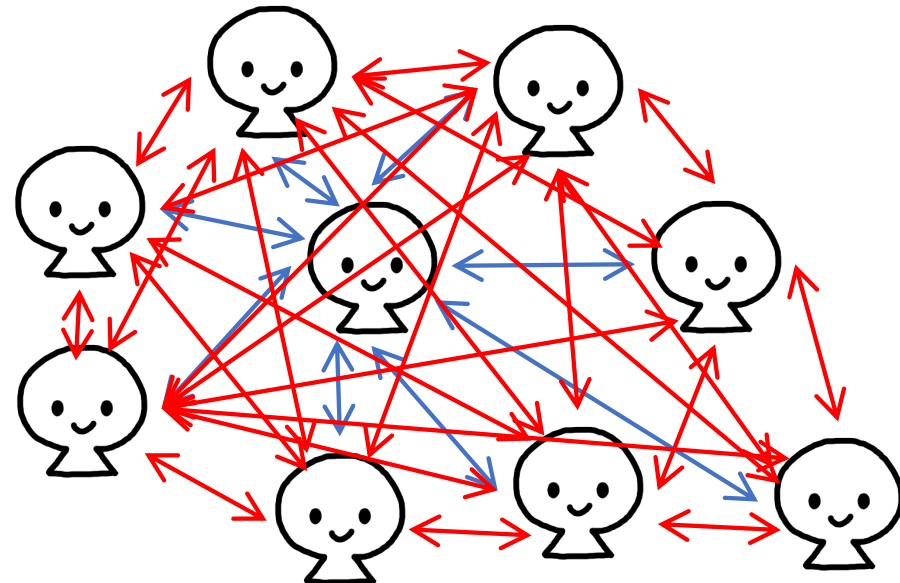
2人 = 1/1本 = カバー率100%



3人 = 2/3本 = カバー率67%



5人 = 4/10本 = カバー率40%

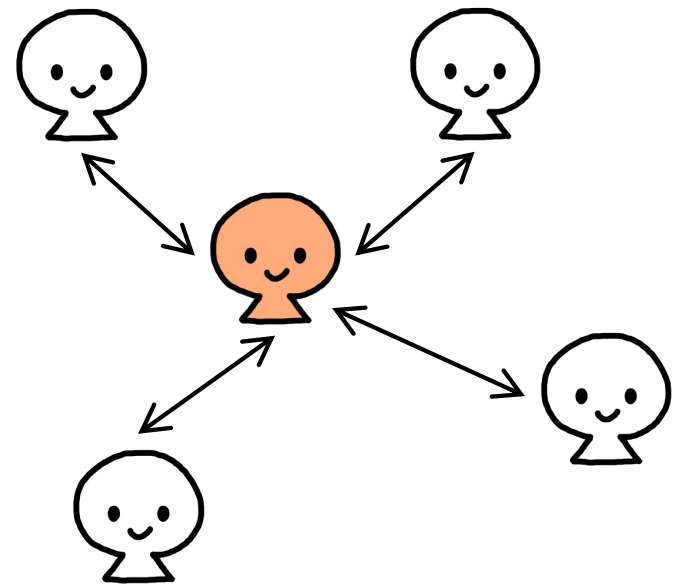
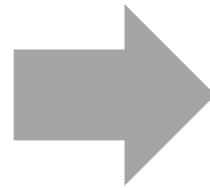
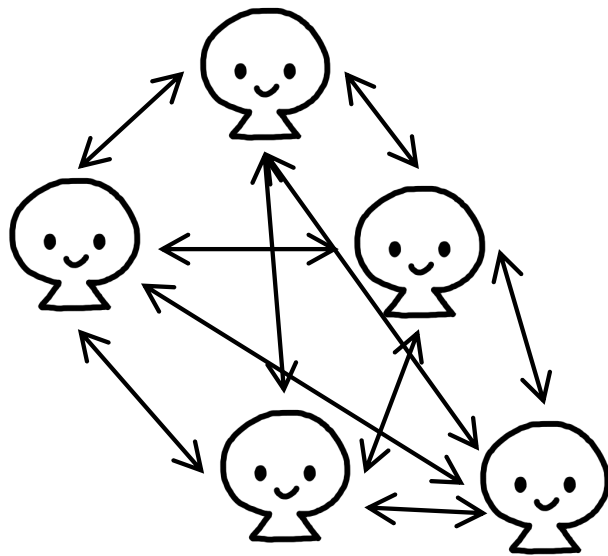


9人 = 8/36本 = カバー率22%

関係者や作業量や取引量が増えるほど情報のやり取りの数が膨大になり、割とあつという間にカオス化して手に負えなくなる

肝は「司令塔」！

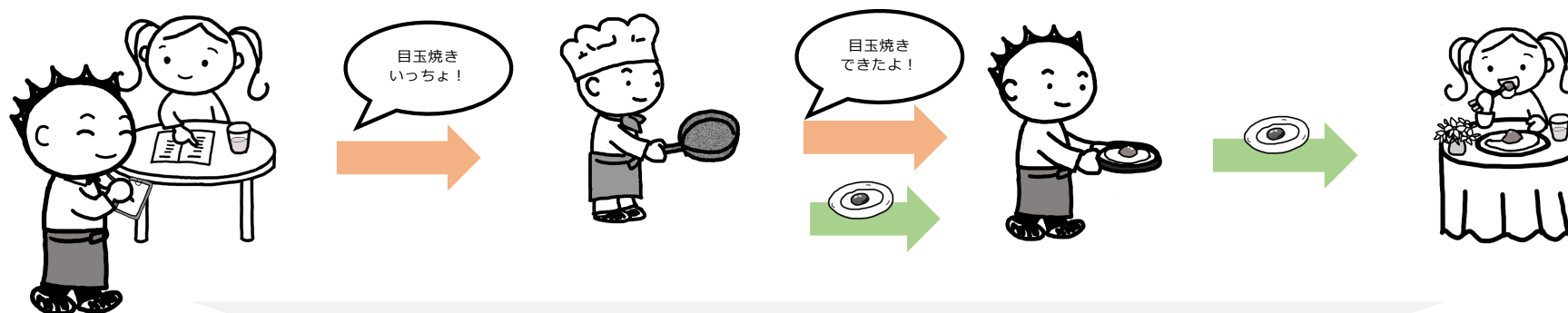
- 誰かに、**全体を俯瞰的に見て采配をふるってほしい**
 - みんな自分の役割で手一杯！
 - 全体の状況を把握して、適切な判断をして、適切に指示してほしい
 - 自分の仕事に集中させてほしい！
- つまり **「司令塔」がほしい！** 司令塔に仕切ってほしい！



コミュニケーションを統括してほしい

司令塔ありきで業務をデザインする！

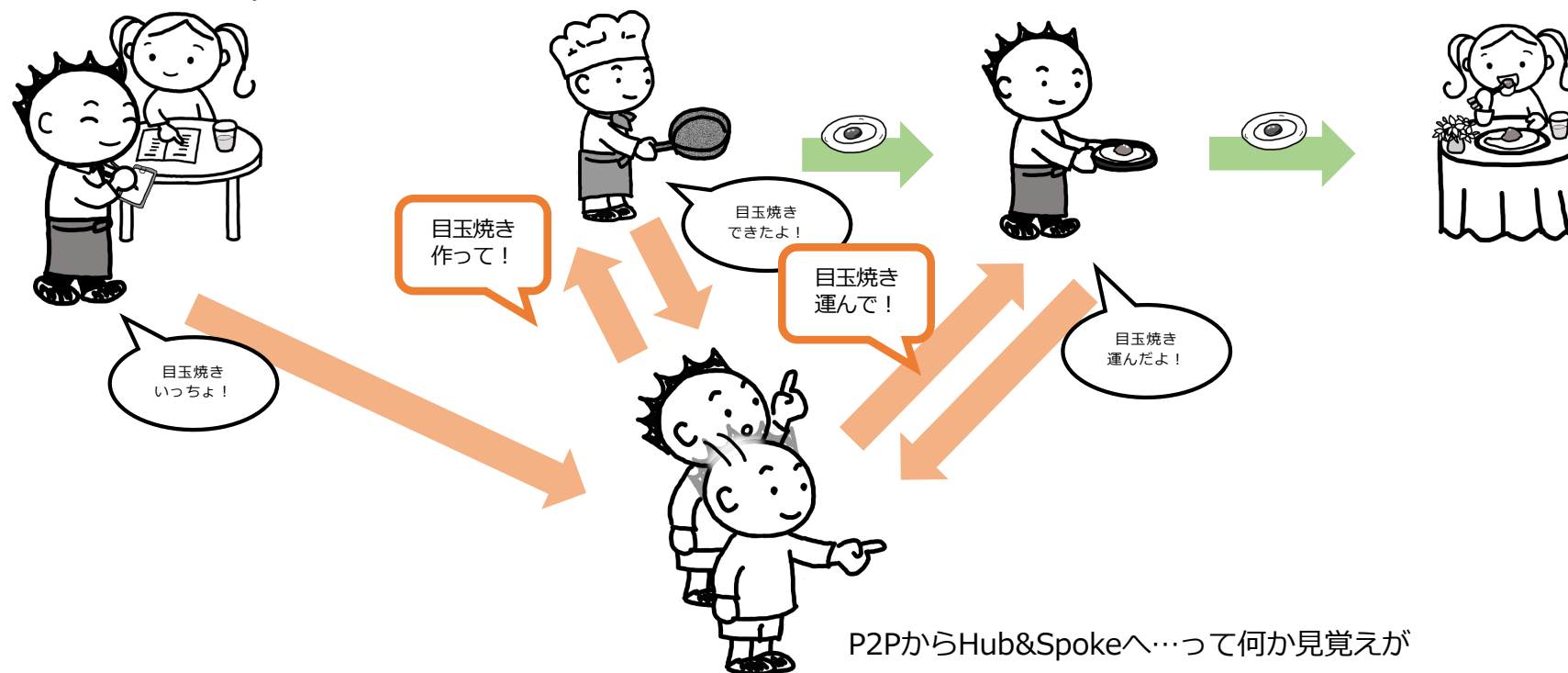
従来の業務フロー



司令塔と各プレイヤーの間のメッセージングを踏まえて業務を考える

メッセージ指向の業務デザイン！

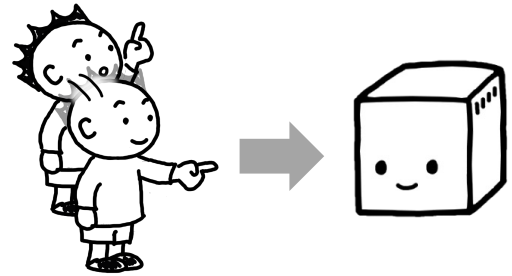
司令塔ありきの業務フロー



P2PからHub&Spokeへ…って何か見覚えが

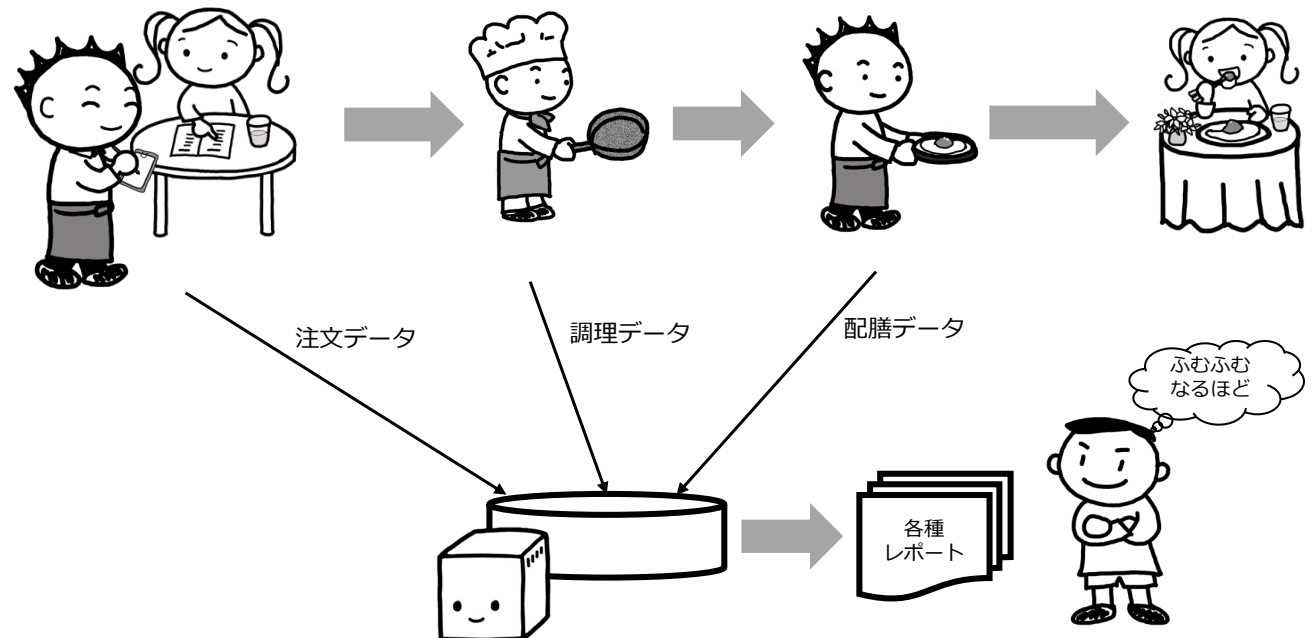
司令塔のデジタル化

- とはいえ現実的には、司令塔役に人手は割けない
→そこでシステム化ですよ！



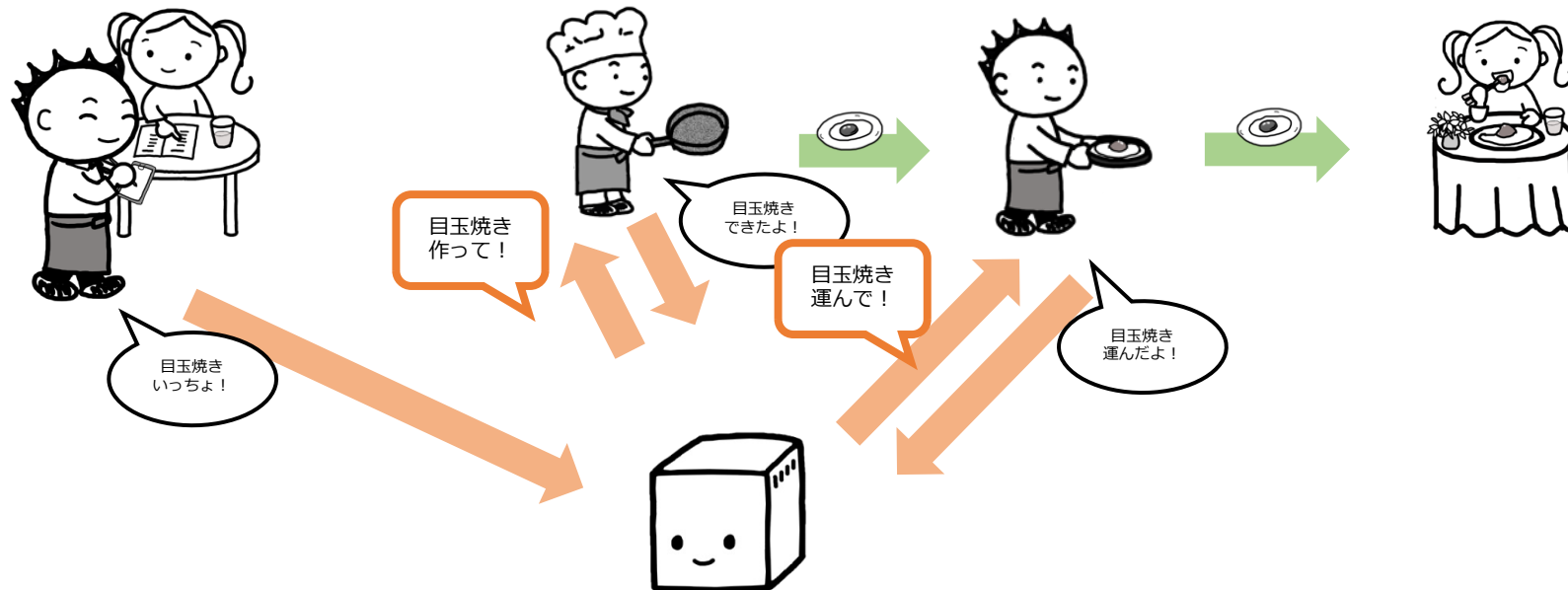
- でも従来の情報システムって

- ぶっちゃけ現場の業務の役に立つというよりも、
管理業務のためでしかなくね？
- システム化しても現場は入力と操作の手数が増える一方で、
全然楽になってる気がしないとかさ



司令塔としてのシステム！

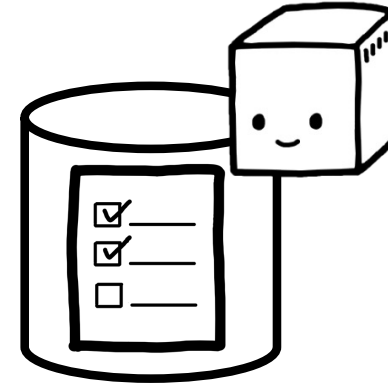
- 事後記録とレポーティングのためのシステム(System of Record)からリアルタイムに関係を扱うシステム(System of Engagement)へ



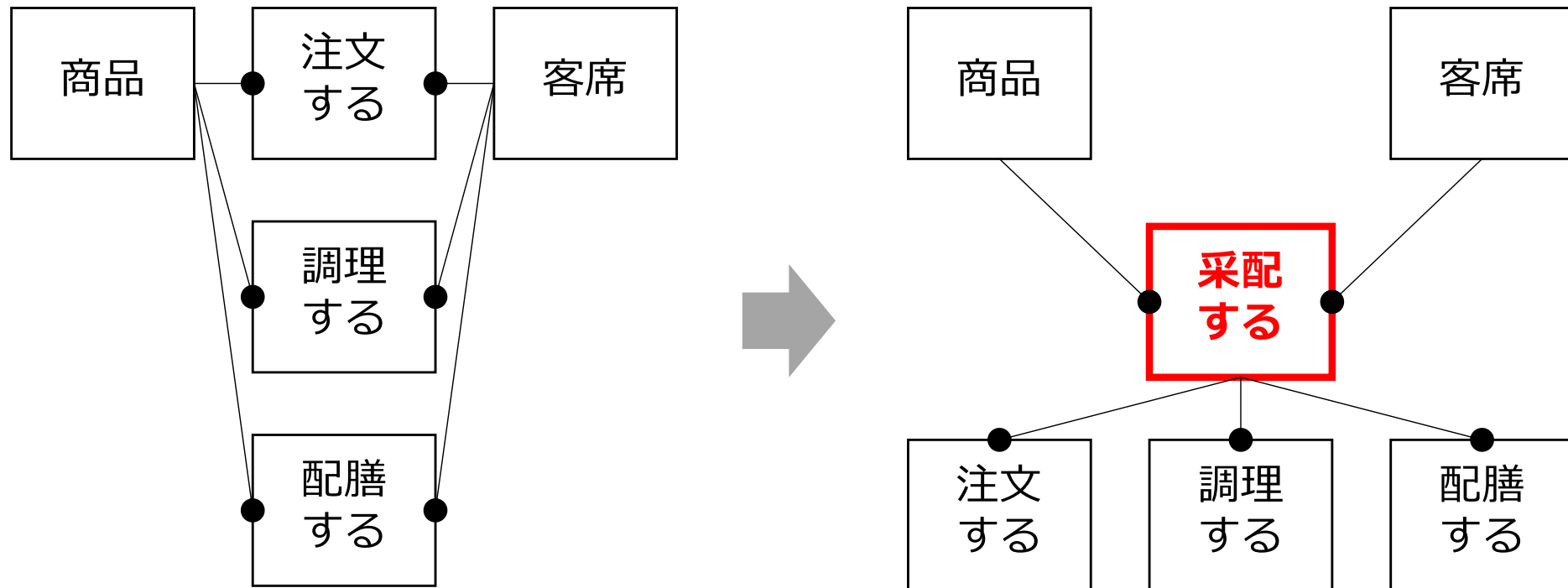
- システム = プログラムの集合体
 - プログラム = データ構造 + アルゴリズム
- 【重要】アルゴリズムはデータ構造に従属する！
 - データ構造つまりデータモデルが肝！
- 司令塔であるシステムが行う「采配する」という業務をデータモデル化する！

業務のデジタルツインとしてのデータモデル

- 司令塔が把握すべき情報
= **コラボレーションステート** (協業の状態)
- コラボレーションステートの構成要素
 - PlanとしてのToDoリスト
(何をどうするのか)
 - 発動条件ルール
(何がどうなったらToDoを行うのか)
 - 司令塔のDoとしての「依頼／指示」
(「誰某さん、このToDoを行ってください」)
 - ToDoが完了したことを表すチェックリスト
(ToDoを完了した"誰某さん"からの完了報告を受けてチェックをつける)



コラボレーションステートを含めたデータモデル



「采配する」の正しいエンティティ名は？

→ 「みんなで連携して行う一連の仕事の集まり」の業務名！

…言い換えると「コラボレーションそのものに名前がついてない」問題
(実はこれ「プロジェクト」や「案件管理」と根っこは同じ概念なんだけど…)

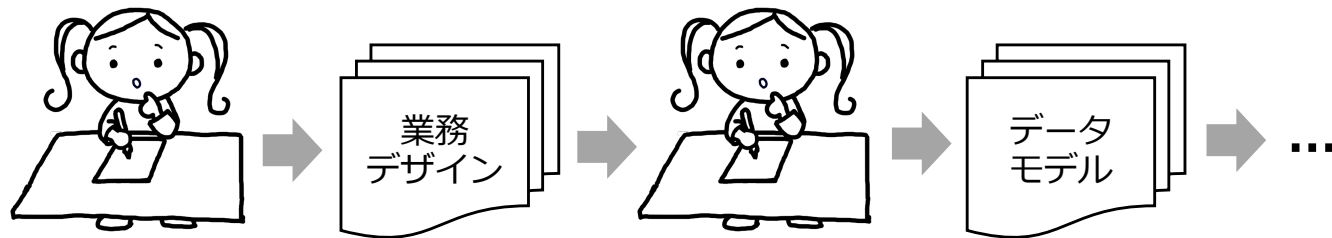
これを意識して考えることが「メッセージ指向の業務デザイン」の要諦！

そしてWagby !

- データモデルさえ作ればWagbyの出番ですよ！
 - データモデル+ルールモデルでシステムを構築できる！
- Wagbyならメッセージングをルールとして制御できる！
 - データが更新されたら通知（メールなど）ができる
 - 時間指定ジョブと実行条件指定がある
- しかもその定義 = 設計情報を
組織のDNAとして引き継ぎ続けることができる！
 - 一般的に、データモデルの寿命はプロセスモデルよりも長い

データモデリングのインプットとしての業務デザイン

- データモデルを作るのもひとつの業務
- 業務にはインプットが必要 = 「業務デザイン」という材料



- 従来のままの業務デザインでは同じことの繰り返しになる
- システムを司令塔に据えて
メッセージ指向で業務を考えることがDXにつながる！
 - メッセージ指向 = 「デマンド（要求）のエクステンションを実現する」

まとめ

- 地に足のついたDX =

**仕事のストレスである
コミュニケーションの問題を
システムで解決する！**

- そのためにまずは「**メッセージ指向の業務デザイン**」！
- 実現のために**Wagby**を！