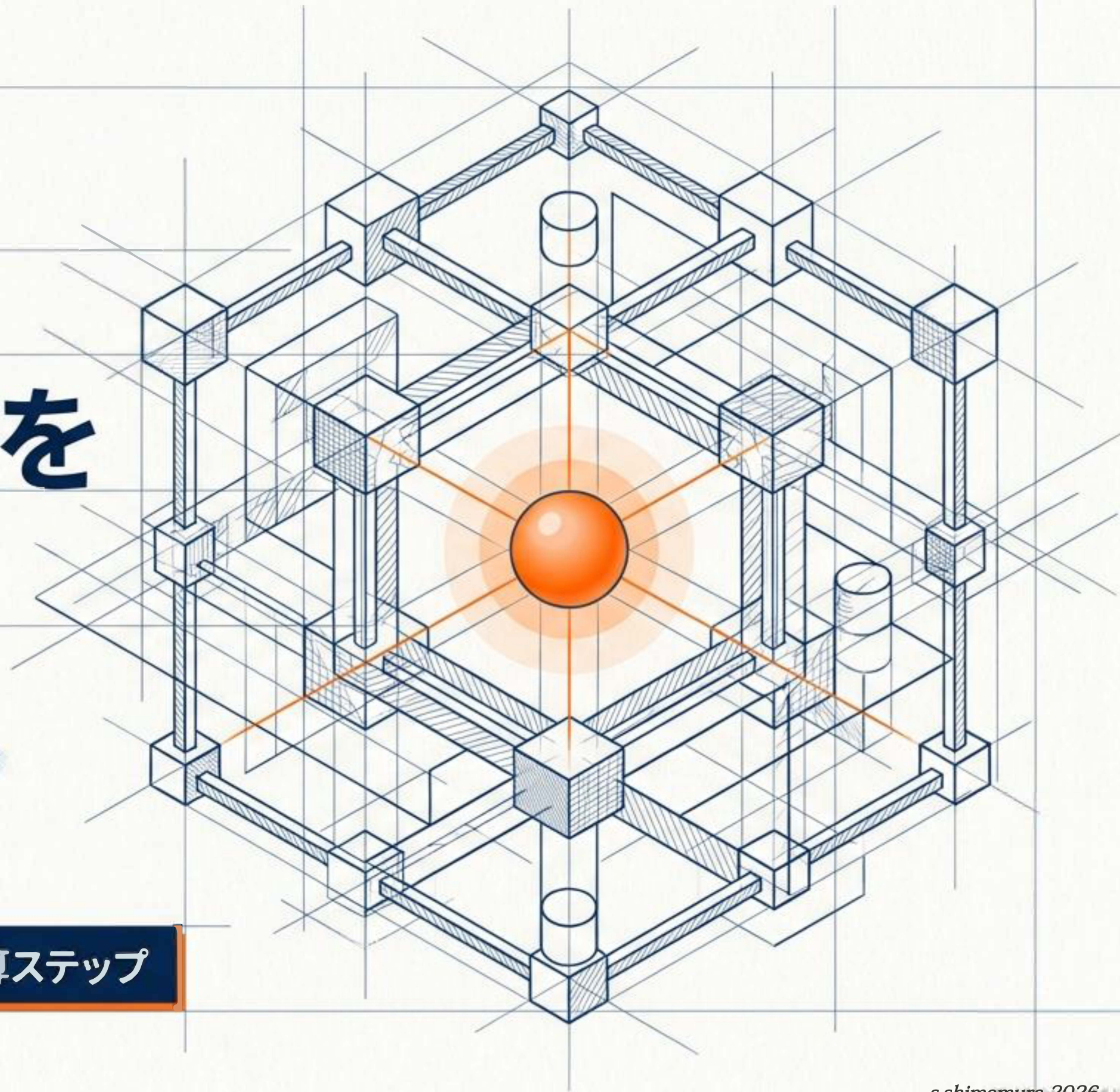


個別支援から 「地域の仕組み」を 設計する

相談支援専門員のための

システム指向ガイド・プレイブック

現場発で制度を動かす4つのスキルと7つの逆算ステップ



個別支援の無限ループに陥っていませんか？

目の前の「個人の困りごと」を解決しても、地域の受け皿がなければ同じ課題が必ず繰り返される。

「通所先が見つからない」

「既存の制度では対応できないニーズがある」

「事業所に受け入れを拒否された」

必要なのは、支援を頑張ることではなく、構造のズレを見抜くこと。

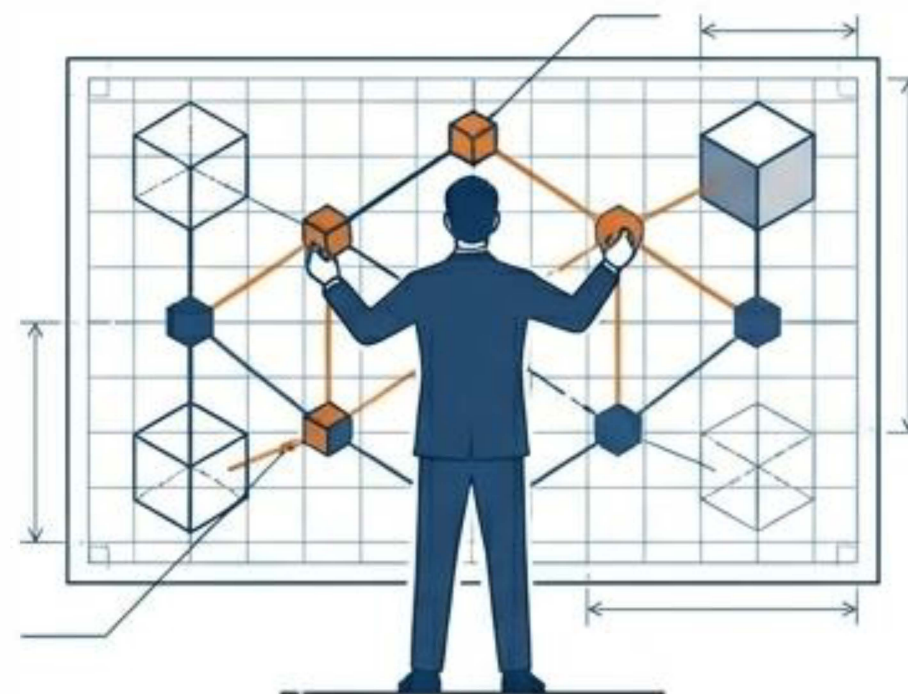
役割の再定義：ケースワーカーから「システムアーキテクト」へ

[Traditional] 個別支援の視点



- 「この人が困っている」
- 支援を調整し、サービスを当てはめる
- 自立支援協議会を「報告と要望の場」とする

[Architect] 地域デザインの視点



- 「同様のケースがあり、〇〇の仕組みが不足している」
- 利害を編み直し、新たな場をつくる
- 自立支援協議会を「解決プロセスを回す場」とする

地域づくりを機能させる鍵は、「個別支援を構造化し、仕組みに変える力」。

地域を変える「変革のエンジン」：4つのコアスキル

1. 翻訳力 (Translation)

課題を言語化する
(個人の悩み → 構造課題)

2. コーディネート力 (Coordination)

仲間を増やす
(利害と意味を編み直す)

4. エビデンス化 (Evidence)

説得力を持たせる
(データ+ストーリー)

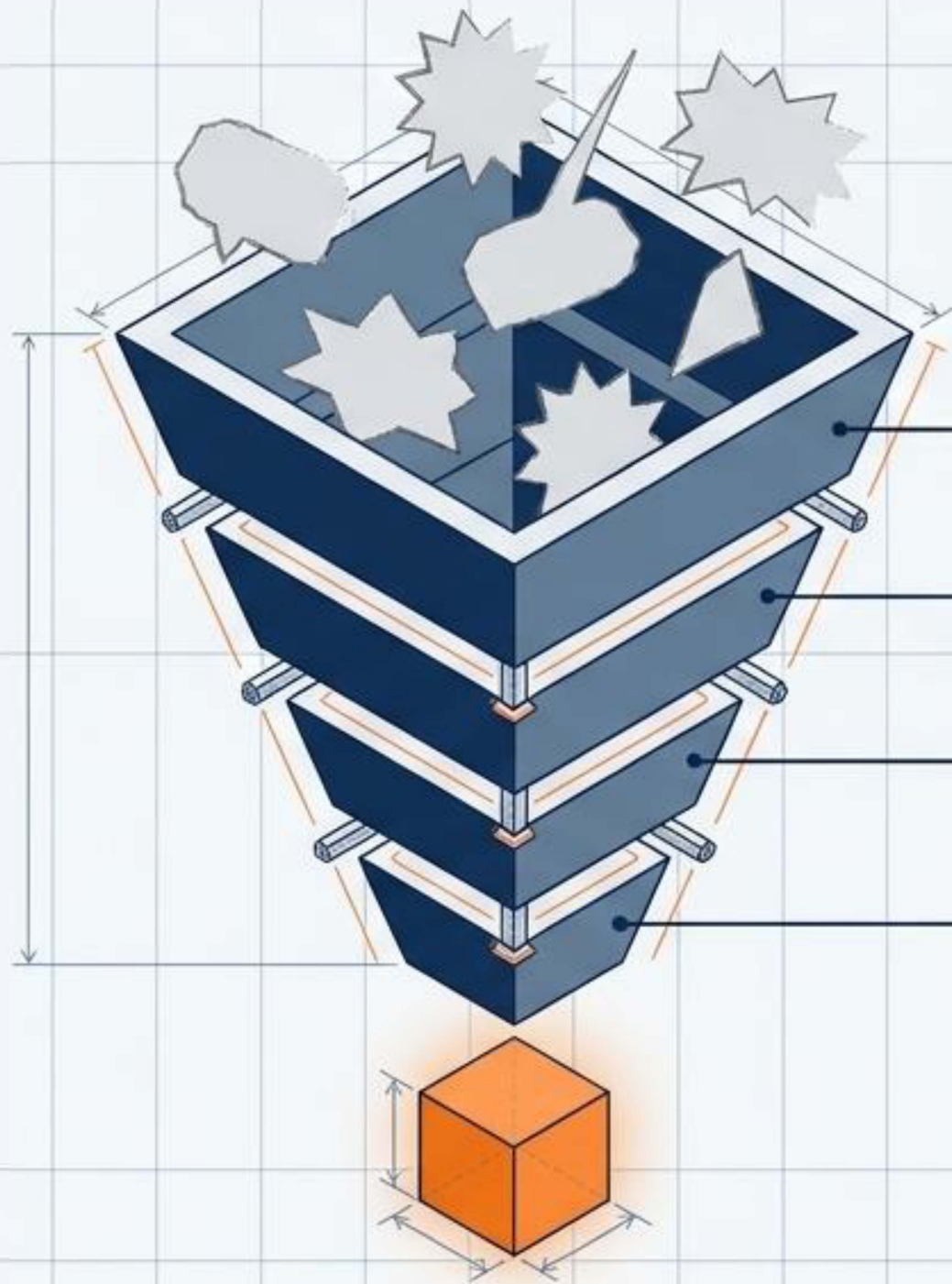
3. プロトタイプ思考 (Prototype)

実践で示す
(小さく試して必要性を可視化)

制度化

この4つが連動することで、初めて「現場発」で地域の仕組みが動く。特に差がつくのは「翻訳力」と「プロトタイプ思考」。

Skill 1: 個人の悩みを構造に変える「翻訳力」



事実 (What)

何が起きているか？

(例: 発達障害の若者が就労移行に定着できない)

背景 (Why)

なぜ起きているか？

(例: 支援密度・負荷が本人の状態と合っていない)

共通性 (Pattern)

他にもあるか？

(例: 複数ケースで同様の離脱が発生)

構造 (System)

制度のどこがボトルネックか？

(例: 軽度～中度の間で、短時間・低負荷から始められる段階的就労の仕組みが欠けている)

【テクニック】 記録段階から「構造メモ欄」を作り、個人名を消して説明できるか自問する。

Skill 2: 利害と意味を編み直す「コーディネータカ」

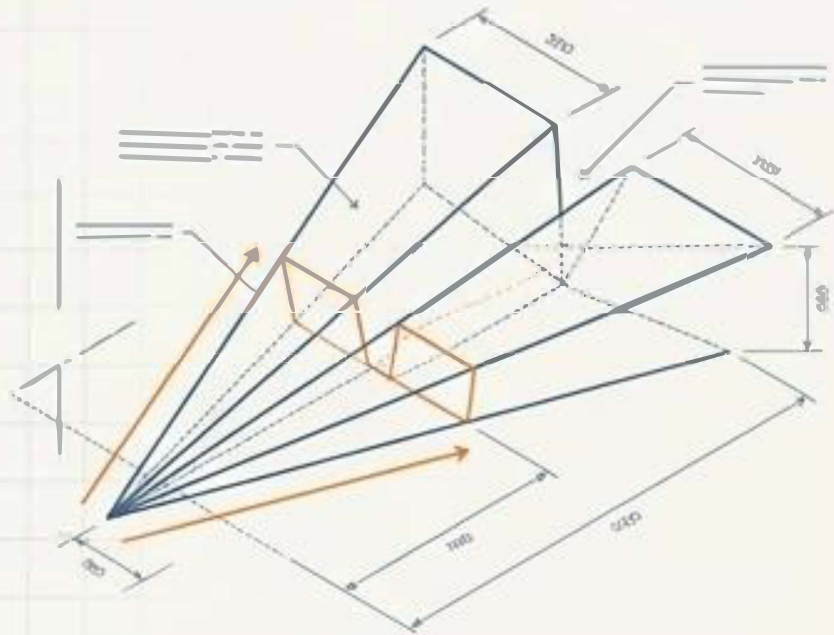
 Architect

主体 (Stakeholder)	暗黙の動機 (Implicit Motive)	提案の翻訳例：就労困難者の受入 (Value Proposition)
行政 (Govt)	制度運用・公平性	「既存制度の隙間を埋めるモデルケースになります」
事業所 (Provider)	人員・収益・リスク	「加算取得や職員研修の仕組み化に繋がります」
医療 (Medical)	安全・治療・防衛	「重度化を予防し、医療費・対応負荷を軽減します」
企業 (Corporate)	生産性・負担軽減	「単なる協力ではなく、人手不足解消と低負担業務の切り出しです」
地域住民 (Residents)	安心・関係性	「孤立を防ぎ、地域の安全網を強化します」

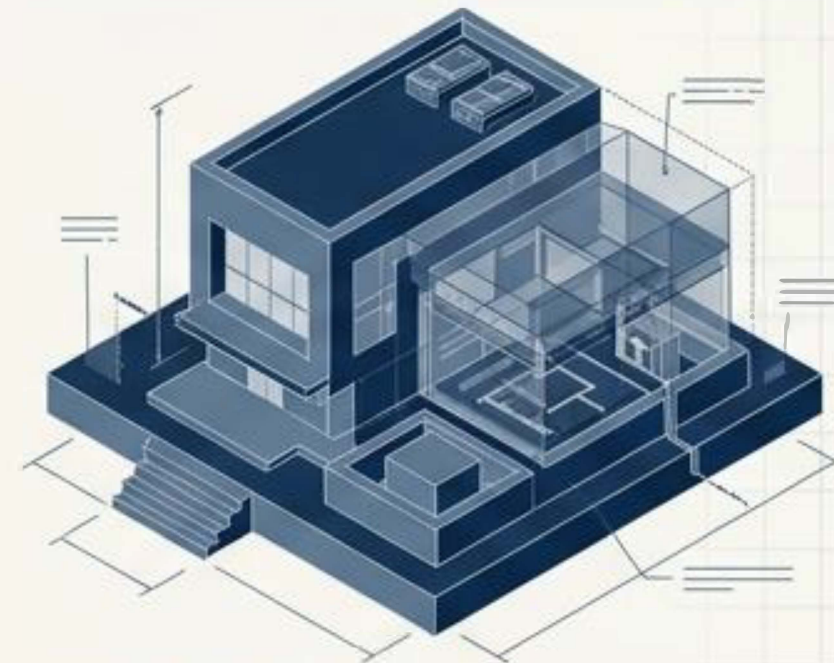
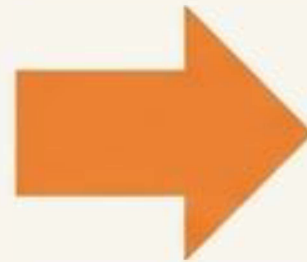
**【鉄則】 「支援をお願いする」のではなく、
「相手にとってのメリット」に翻訳して巻き込む。**

Skill 3: 小さく試して見せる「プロトタイプ思考」

地域は前例主義。構想だけでは動かない。「机上の課題」を「実証された課題」に変換する。





プロトタイプ (小さく試す)



制度化 (常設の仕組み)

プロトタイプ設計の3原則

- ✓ 小さい: 低コスト・既存資源の流用 (失敗してもいい設計) 
- ✓ 早い: 短期間で実施 (例: いきなり常設拠点ではなく、月1回のサロンから)
- ✓ 測れる: 写真や記録を残し、客観的に評価可能にする 

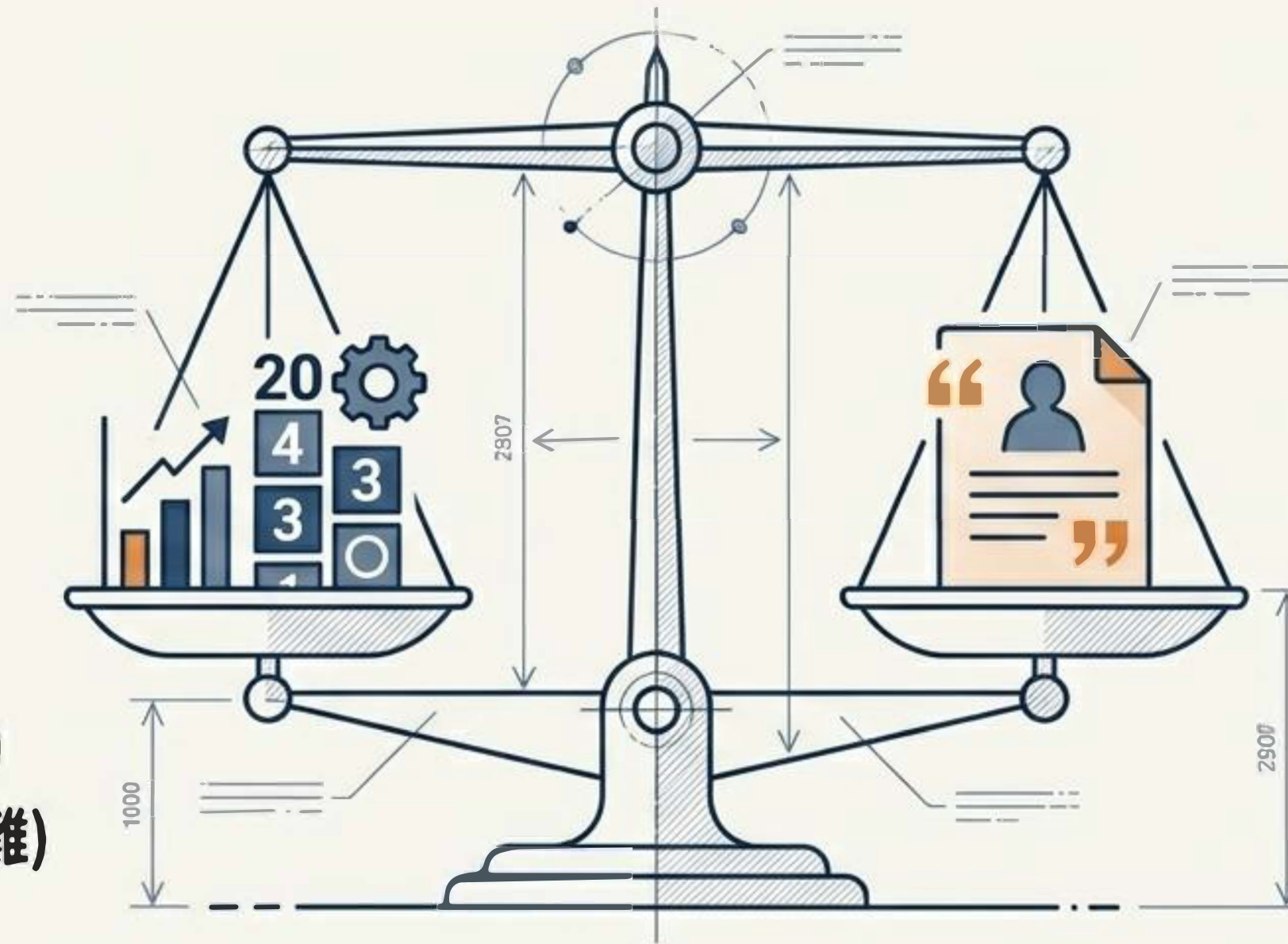
設計フォーマット例

【目的】何を検証するか	<input type="text"/>
【対象】誰か	<input type="text"/>
【方法】どうやるか	<input type="text"/>
【期間】いつまで	<input type="text"/>
【指標】何を測るか	<input type="text"/>

Skill 4: 実感を説得力に変換する「エビデンス化」

定量データ (頭を動かす)

- 件数
(年間20件の発生)
- 支援負荷
(平均調整期間3ヶ月)
- 結果 (○件が解決困難)



定性ストーリー (心を動かす)

- 「このままだと母親が介護離職寸前になる」という具体事例1つ

【WARNING】 データだけでは心が動かない。事例だけでは説得力が弱い (制度化の根拠にならない)。「10件中7件が○○+具体事例1つ」のセット提示が鍵。

自立支援協議会の再定義：出口ではなく「設計条件」

協議会は放っておくと「報告の場」になる。現場側がどこまで逆算して設計して持ち込むかが全て。

PATHWAYS

PATH A: THE TRADITIONAL DEAD END

個別支援を
頑張る

課題が限界まで
溜まる

協議会で
「困っています」
と報告

[行き止まり:
具体的な動きなし]

個別のケース
(Case)

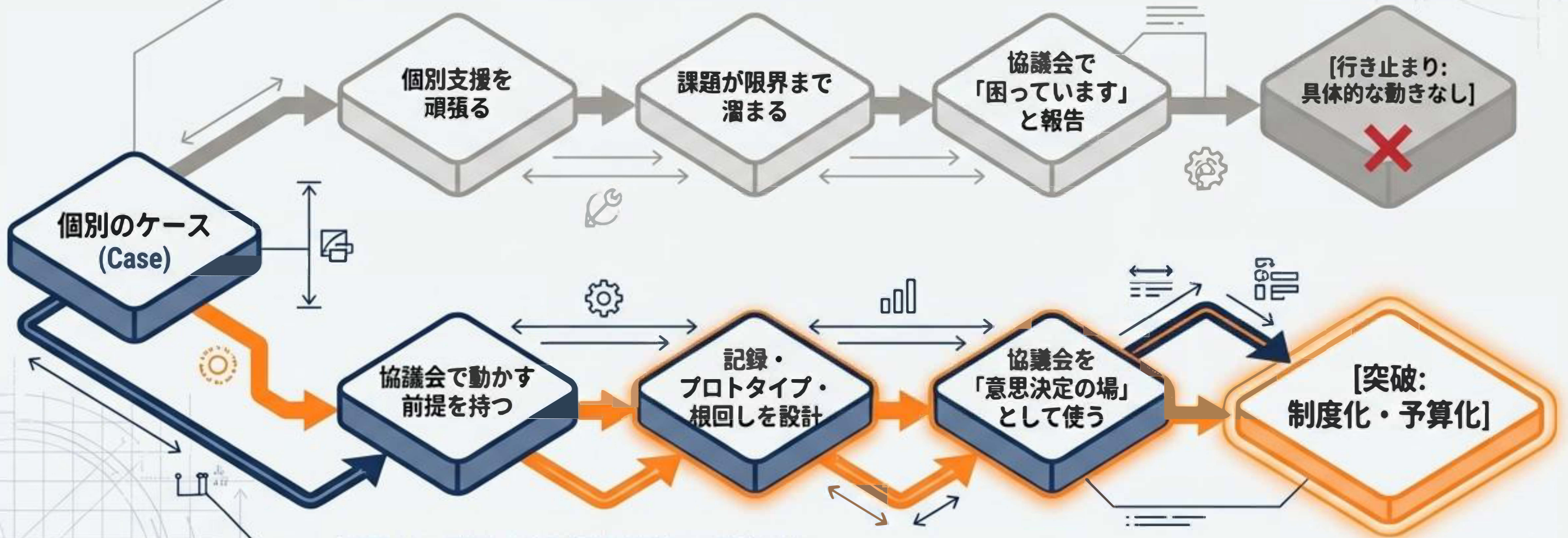
協議会で動かす
前提を持つ

記録・
プロトタイプ・
根回しを設計

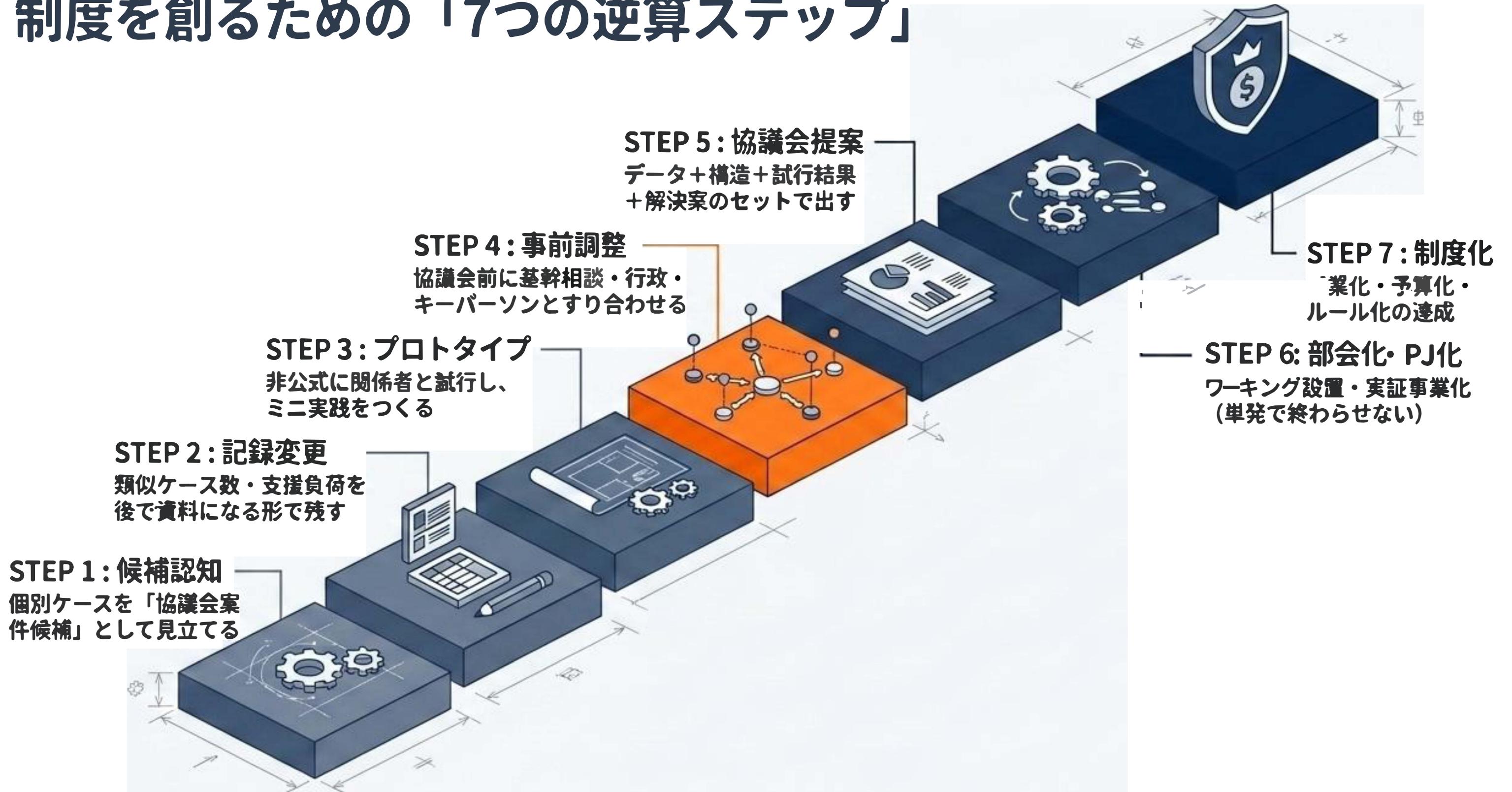
協議会を
「意思決定の場」
として使う

[突破:
制度化・予算化]

PATH B: THE ARCHITECT'S HIGHWAY



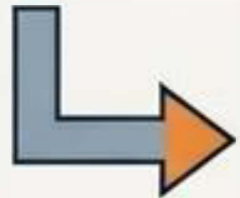
制度を創るための「7つの逆算ステップ」



システムアーキテクトが避けるべき「4つの罠」



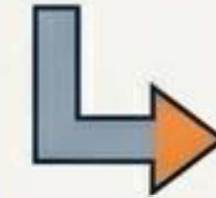
罠1：相談員だけで抱え込む



【対策】早期の外部化。一人で解決しようとせず、初期段階から行政や基幹相談と「**共通の悩み**」として共有する。



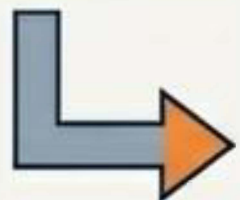
罠2：正義感で突っ込む（行政批判）



【対策】構造にフォーカス。「**制度が悪い**」と攻撃すれば反発される。「**現行制度に隙間がある**」と客観的課題に変換する。



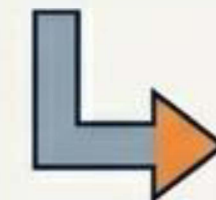
罠3：課題が大きすぎる・抽象的



【対策】解決可能なサイズへの分解。「**地域の受け皿づくり**」ではなく、「**まずは週1回の受入モデル試行**」から始める。

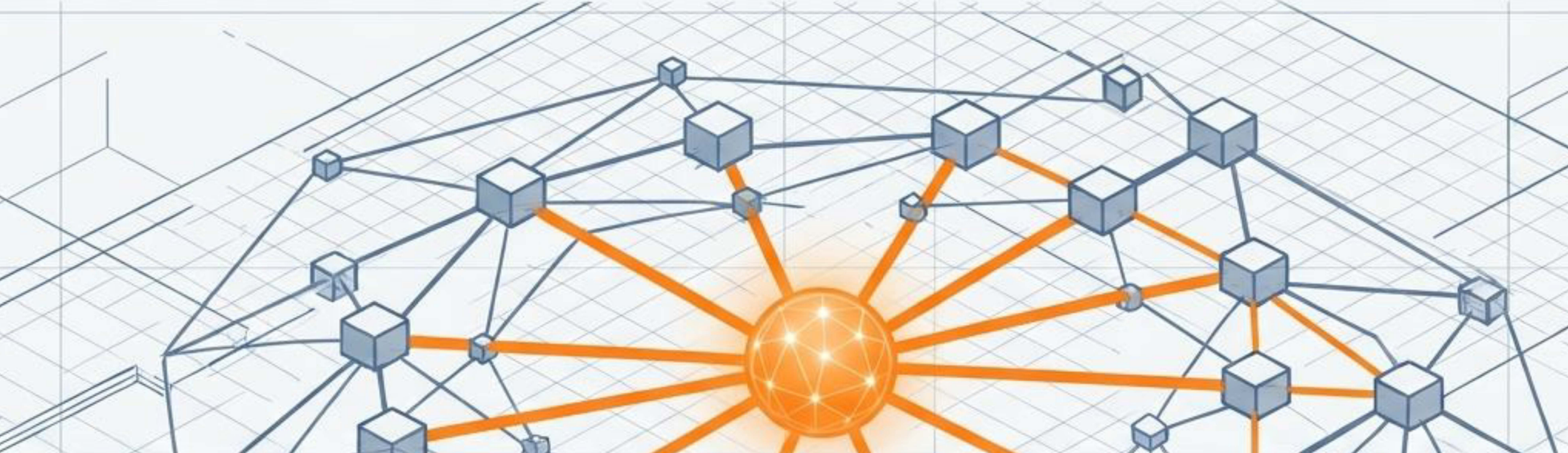


罠4：打ち上げ花火で終わる



【対策】時間軸と担当の固定。単発の議題提起で満足せず、「**部会化やワーキンググループ設置**までを**ゴール**に設定する。

現場発で、地域の仕組みを動かす人へ

- 
- 自立支援協議会は、与えられた場ではありません。
 - 使い方を変えれば、地域を変える最大の「エンジン」になります。
 - 違いはただ一つ。「現場側がどこまで逆算で設計して持ち込むか」です。

個別の支援から、地域全体のデザインへ。あなたはもう、**システムアーキテクト**です。