PortX

競争優位を、今日から実装する ユーザー主導SIのためのOS

PortXは、大企業の業務を実装するソフトウェア会社。
Spec-Driven Development プラットフォーム Formula を自社で開発・運用しています。

PortXを一言で

PortXは、大企業の業務を実装する
ソフトウェア会社です。
SI会社ではありません。受託開発会社でもありません。

業務を最も知る人と、ITを最も知る人が別人であるという分離が、エンタープライズSIの歪みの根本原因。
PortXは、その分離を消すための**ソフトウェア(Formula Platform)**を自社で開発・運用し、
大企業のユーザー部門が「業務を変える主導権」を取り戻せる構造をつくっています。

PortX

世界の見方 ①

エンタープライズSIの歪みは、たった1つの分離から派生している

「業務を最もよく知る人」と「ITを最もよく知る人」が別人であること。この一点から、4つの構造歪みが必然的に生まれる。

ROOT CAUSE

業務知識はユーザー企業の現場部門に、IT実装能力はベンダー企業の末端エンジニアに偏在している。この2つが直接つながらないから、間に多重下請け・要件定義書・仕様書・テスト仕様書という「翻訳レイヤー」が積み重なり、歪みが固着する。

派生する歪み 1 — 価値と対価の分離

業務を最も知る末端エンジニアの単価が、最も低い

業務を翻訳して伝える階層が必要なため、元請→二次→三次→末端の多重下請けが成立する。1人月150万→30万まで中抜きされ、**業務知識に最も近い人ほど報酬が低い**逆ピラミッド構造になる。

派生する歪み 2 — 効率化と売上の逆相関

生産性を上げるほど、ベンダーは貧しくなる

業務とITが分離していると価値の単位が「機能」になり、必然的に「人月」で計らざるを得ない。**人月で売る限り、効率化するほど請求額が減る**。業界全体が生産性を上げないインセンティブに縛られる。

派生する歪み 3 — ユーザー部門の主導権喪失

業務で困っている人が、最後まで発注に届かない

分離があるから情シスという翻訳層が必要になる。予算・発注権が情シス経由になり、**現場の痛みは情シスを経由するうちに二次情報化**し、発注書に辿り着く前に変質する。

派生する歪み 4 — 仕様書と実装の構造的乖離

同じ業務が、5フェーズで5つの別書式に作り直される

業務とITの間に翻訳が必要だから、要件定義書→基本設計書→詳細設計書→実装→テスト仕様書という**同じ情報の多重コピー**が成立する。後半になるほどズレの影響が膨らみ、手戻りコストが爆発する。

※ この構造を、経済産業省自身が **"ユーザー企業とベンダー企業の低位安定の共依存"** と公式に名付けている（DXレポート2.1、2021年）。意思の問題ではなく、仕組みそのものの問題であることは国の認識でもある。

世界の見方 ② — 派生する歪み 4 のディープダイブ

SIは「同じ業務情報」を、5フェーズで何度も作り直し続けている

業務とITが分離している以上、その間を翻訳する大量の中間成果物が必要になる。**同じ情報がフェーズごとに別書式で再生成される**のがSIの本質的な非効率。

PHASE 1	PHASE 2	PHASE 3	PHASE 4	PHASE 5
要件定義	**設計**	**開発**	**テスト**	**運用・保守**
・業務フロー図 ・機能一覧 ・画面一覧 ・要件定義書	・画面設計書 ・API設計書 ・テーブル定義書 ・処理設計書	・画面コード ・API実装 ・DB構築 ・処理ロジック	・テスト仕様書 ・テスト実行 ・エビデンス作成 ・障害管理	・運用手順書 ・保守資料 ・変更管理書 ・障害対応書

1. 情報の揺れ

同じ業務情報がフェーズごとに別書式に翻訳される。**翻訳のたびにズレが入り、後半になるほど蓄積**する。

2. 手戻りの爆発

テスト段階でズレが発覚すると、修正コストは要件段階の**数十倍**に膨れる。
やり直しは全フェーズに波及する。

3. 膨大な人手と時間

作るのも、レビューするのも、修正するのも、すべて人手。**SIプロジェクトの工数の大半**がこの「情報の繰り返し生成」に消える。

PortXのアプローチは「AIで速くコードを書こう」ではない。

この「同じ情報を別書式で何度も作り直す」構造そのものを、業務仕様書を唯一の正本(SSOT)に置くことで消す。**差は「速さ」ではなく「決定論性 (determinism)」**。Vibe coding (V0/Bolt/Lovable/Cursor) は確率的 = 動くこともある。Spec-Driven Development は決定論的 = 毎回正しく動く。本番稼働するエンタープライズSIは、後者でしか作れない。

PortX

世界の見方 ③ — 構造固着の定量化

この歪みの大きさは、国の公式数字でも見えている

前ページの分離構造を、別の角度から定量化したもの。全て公的機関の一次情報。

2025年の崖

12兆円

レガシーシステムを放置した場合、2025年以降に発生する**年間経済損失**。経産省DXレポート2018年の警告。2025年5月の公式レポートで問題の恒常化を追認。

IT予算の使途

8割

既存システムの維持運用に消えている。新規開発・DX投資に回るのはわずか2割。JUAS調査では2013年79.1%→2018年77.5%と5年でほぼ改善なし。

IT人材不足

79万人

2030年の**IT人材不足見通し**（経産省高位ケース）。既に2025年時点で43万人が不足。人を増やすアプローチでは構造的に解けない。

レガシー残存率

61%

ユーザー企業の61%にレガシーシステムが残存。大企業に限れば**74%**。経産省2025年5月公式レポート。

製造業IT投資比率

1.0% vs 3.3%

日本の製造業の売上高IT比率は1.0%。**米国（3.3%）の1/3**しかなく、構造的な投資不足が継続（NRI / Gartner）。

PortX

Why Now

大企業を生んだのは、いつも「指数関数的な追い風」だった

マイクロコンピュータ→Intel・MS・Apple、インターネット→Google。**大企業は意思決定の速さでは作れず、テクノロジーの指数関数的変化を追い風にする以外にない**。今、ソフトウェア生産性そのものに、その変化が起きている。

PAST INFLECTIONS

過去の指数関数的変化と、それを掴んだ企業

1970s 半導体の集積密度 → **Intel**

1980s パソコン処理性能 → **Microsoft / Apple**

1990s インターネット接続性 → **Google / Amazon**

2000s モバイル普及率 → **Apple iPhone**

2010s クラウド計算量 → **AWS / Salesforce**

どの時代の大企業も、新しいインフラが標準化する直前の**数年**に立ち上がっている。

CURRENT INFLECTION (2023〜)

LLMによるソフトウェア生産性の指数関数的向上

要件整理・設計・実装・テスト・変更管理 — SIの「知的作業」の大半が、わずか2〜3年でAIに代替可能になった。人月で人が手作業していた工程の前提が、根本から崩れている。

このインフラが「当たり前」になる前に、仕組みを根本から作り変えたプレイヤーだけが、次の大企業になる。

なぜ「今」でないとダメか：5年後、LLMによる開発生産性向上は全SIerに当たり前のものとして普及する。その時、**「人月」を前提に組まれた既存SIerの組織・契約・原価構造**は、AI前提で組まれた新しいプレイヤーに置き換わる。**置き換わる側に立てる窓は、今後数年しか開いていない**。

PortXのソリューション

Formula = エンタープライズSIのための Spec-Driven Development プラットフォーム。

業務仕様書を唯一の正本(SSOT)として、画面・DB・API・テスト・運用までを派生させ続ける**「フルスクラッチのようなSaaS」**。
新規構築も追加開発も運用保守も、すべて**同じパイプラインの上で決定論的(deterministic)に動く**。

VIBE CODING (V0 / Bolt / Cursor / Lovable)

確率的 — Might work

プロンプト→生成→修正の繰り返し。デモまでは速い。ただし**エッジケースで壊れ、変更が予期せぬバグを生み、本番で破綻**する。

SPEC-DRIVEN DEVELOPMENT (Formula)

決定論的 — Does work

業務の仕様書を先に書き、AIがその仕様通りに成果物を生成・検証。毎回同じ振る舞いが保証される本番品質に到達できる唯一の道。

差は「速さ」ではなく「決定論性 (determinism)」。AIはソフトウェアを動かすものを変えていない。**最も大事な工程 = 仕様書を、書きやすくしただけ**。

PortX

Formulaのアーキテクチャ

新規構築も、運用後の追加開発も、同じ1つの仕組みで動く

人が書き換えるのは業務の仕様書ただ1つ。そこから全ての成果物が生成・同期される。**この仕組みは構築フェーズでも運用フェーズでも変わらない。**

唯一の正本

業務の仕様書

人が書き換えるのは
これだけ

変更方法

日本語で指示

「検索条件を追加して」

「タブを4つに分けて」

仕様書から自動で生成されるもの

画面
ブラウザで動くユーザー画面

データ処理
業務ロジック・計算・連携

データベース
テーブル定義＋初期データ

変更履歴
変更内容・影響範囲の自動記録

自動検証
画面・処理・データの動作確認

エビデンス
スクリーンショット＋検証レポート

速さ

圧倒的に速い

成果物を人手で作り直す工程がゼロ。仕様変更から全成果物の更新までが即座に完了。

品質

構造的に高品質

正本から自動生成するため「設計書と実装が違う」「テスト漏れ」が構造的に起きない。

コスト

コスト構造が違う

設計書作成・レビュー・転記・テスト作成。従来の工数の大半をAIが自動で処理。

Formulaのパイプライン構造

業務仕様書から本番システムまで、5段で自動製造する

Formulaは「コードを書くツール」ではなく、業務仕様書を入れたら本番システムが出てくる**自動製造ライン**。新規構築でも運用後の追加開発でも、同じ5段が動く。

STAGE 1

/workflow

業務フローの構造化

FDEのヒアリング内容を、AIが業務フロー(誰が・何を・どの順で)に構造化する。

OUT: workflow.md

STAGE 2

/generate

仕様書から成果物を自動生成

spec(view/entity/api)から、画面・データ処理・DB・テストを自動生成。

OUT: .svelte / .ts / .sql

STAGE 3

/change

変更要求を構造化(Delta)

「検索条件に部署を追加」等の変更を、intent / changes / impacts のYAMLに変換。

OUT: delta.yaml

STAGE 4

/sync

全成果物に一括反映+整合性保証

Delta群を読み、画面・処理・DB・テストに同時反映。view/api/entityのカラム整合性を機械的に検証。

OUT: 更新済 全成果物

STAGE 5

/evidence

テスト実行+エビデンス生成

テスト自動実行、スクリーンショット取得、検証レポート出力。本番リリースの根拠が機械的に揃う。

OUT: テスト結果 / 監査証跡

INPUT （人間が書くもの）

業務仕様書 (workflow / view / entity / api)

人間が書き換えるのはここだけ。日本語の指示も可。

PIPELINE （AIが回すもの）

5段の自動製造ライン + Skill群

17以上のSkill (業務知識化された手順) が各段で動く。

OUTPUT （本番に出るもの）

本番稼働可能なシステム + 監査証跡

プロトタイプではなく、運用・保守・変更に耐える本番品質。

構築フェーズも運用フェーズも、流れるのはこの5段だけ。同じ仕様書を変えて再度パイプラインを回せば、全成果物が整合性を保ったまま更新される。**これが「フルスクラッチのようなSaaS」の中身。**

PortXの差別化 ① — 組織×プラットフォーム

人間は競争優位の設計に。AIは実装の自動製造に。

PortXはFormulaを売る会社ではない。**FDEという組織能力と、Formulaという自動製造基盤が密結合した「事業モデル」**そのものが差別化の本質。

人間にしかできない領域

競争優位の設計

Forward Deployed Engineer（FDE）

顧客の現場に常駐し、業務を深く理解する人間エンジニア。Palantirが体系化したモデルをPortXが日本の製造業向けに進化させたもの。

FDEが担う4つの知的作業

① 顧客業務のヒアリングと構造化

② 現状プロセスの問題特定

③ ありたき姿の設計と効果の刈り取り方

④ 仕様書（spec.md）への落とし込み

AIで代替する領域

システム設計・実装・運用の全工程

Formula

spec.md を唯一の正本（SSOT）として、画面・データ・API・テスト・エビデンスまで全成果物を自動生成するAI製造基盤。

Formulaが担う自動化

① 仕様書から画面・処理・DBを自動生成

② 変更要求をDelta化して全成果物に一括反映

③ テスト実行とエビデンスの自動取得

④ 仕様書と実装を構造的に一致させ続ける

設計判断 — なぜGUIに投資しないか

V0・Bolt・Lovableのような「誰でも画面を作れるツール」が作れるのは、**仕様書のない vibe coding ＝ プロトタイプ止まり**。本番稼働には ① 業務例外・判断ロジックの網羅 ② 変更時の影響範囲の追跡 ③ データ整合性・監査要件 ④ 運用フェーズの保守可能性 が必須で、これらは **業務仕様書を正本(SSOT)に置く構造** でしか担保できない。

PortXが投資しているのは、上流の業務知識を仕様書として固定し、そこから全成果物を派生・同期し続ける **Formula Platform Layer**(spec・Delta・Sync・Skill群)。次ページで詳述する。

PortXの差別化 ② — 技術アーキテクチャ

投資先は GUI でも コード生成 でもなく、Formula Platform Layer。

V0・Cursorはコードを速く書くツール。Formulaは **spec(業務仕様) → Delta(構造化変更) → Sync(整合性保証つき自動反映)** という独自のパイプラインに投資している。

❌ 投資していない層

GUI / UI生成

V0、Bolt、Lovable等が投資している層。**仕様書がないvibe codingではプロトタイプ止まりで本番稼働できない**(整合性・例外処理・運用保守が不可)。PortXは投資対象としない。

❌ 投資していない層

コード補完 / 生成

Copilot、Cursor、Devin等が投資しているレイヤー。個人開発者や小規模案件には強いが、**「動いた、でもなぜこう作ったか残らない」問題**でエンタープライズ保守が成立しない。

✅ PortXが投資している層

Formula Platform Layer (spec × Delta × Sync)

業務の仕様書を正本に、変更をDelta化し、Syncで全成果物に一括反映する**SI業務専用のAIオーケストレーション層**。汎用LLMの上に、エンタープライズ業務特有の制約・整合性ルール・運用フェーズの継続性を実装している。

技術的キモ ①

spec.md = 唯一の正本

workflow.md / view.md / entity.md / api.md の総称。人間が読み書きするのはここだけ。**.svelte / .ts / .sql は全てここから派生**するので、仕様書と実装が構造的に分離しない。

技術的キモ ②

Delta = 変更の構造化

「検索条件に部署を追加して」等の変更要求を**intent / changes / impacts の構造化YAML**に変換し蓄積。変更の意図と影響範囲が常に追跡可能。通常のgit diffにはできない知識資産化が可能に。

技術的キモ ③

Sync = 一括反映と整合性保証

Delta群を読み、**画面・データ処理・DB・テストに同時に反映**。view.mdとapi.mdのカラム整合性を自動チェックし、ERRORなら修正。人間のレビューではなく**機械可読なチェックリスト**で品質保証。

PortXの差別化 ③ — フロンティアモデルとの境界線

Formulaは ツールではなくサービス。だからモデルが強くなるほど強くなる。

If you sell the tool, you are in a race against the model. If you sell the work, every improvement in the model makes your service faster, cheaper, and harder to compete with.

Formulaは「画面を作るツール」ではなく、「業務システムを作り続ける仕事そのもの」を提供する。**フロンティアモデルが強くなるほど、Formulaという仕事は速く・安く・複製困難になる。**

LAYER 1 — 基盤モデル

フロンティアモデル本体

担い手: OpenAI / Anthropic / Google

汎用的な言語理解・コード生成。圧倒的な研究開発投資。**PortXは戦わない**。むしろ全面的に依存し、進化の果実を取り込む側に回る。

LAYER 2 — 汎用開発支援

コード生成 / IDE統合

担い手: Cursor / Copilot / Devin / V0

汎用的なコーディング体験を磨く層。**FMの能力向上で価値が薄まる側**。FMが直接コードを書けるほど、エディタ統合の差は縮まる。

LAYER 3 — ドメイン特化ソフトウェア

エンタープライズ業務ソフトウェア

担い手: PortX Formula

FMの能力を、エンタープライズの中核業務という特殊な制約条件に変換するソフトウェア層。FMが強くなるほど自動製造の精度が上がり、PortXの強みも増幅する。

なぜ FMプロバイダーは Layer 3 に直接降りてこないか

① 顧客の業務に常駐する組織がない

FMプロバイダーは「水平展開」が事業構造。顧客現場に常駐するFDE組織は彼らの強みと真逆。

② 業務SI特有の制約条件がコスト化する

日本の製造業の業務制約・例外処理・情シス慣習等は、研究組織にとっては**学習データになりにくい雑音**。直接吸収する経済性が低い。

③ 失敗しても撤退できない

エンタープライズSIは継続性が命。研究資源を局所最適に張れないFMプロバイダーには、構造的に向かない事業領域。

→ PortXが Layer 3 に投資している限り、Layer 1 の進化は**脅威ではなく追い風**。**Formulaの土台を早く作り始めて磨き続けたプレイヤーほど、後から参入したベンダーにも従来SIにも構造的に勝つ。**

PortXの差別化 ④ — 他プロダクトとの比較

3つのプレイヤーマップと、PortXの位置づけ

AI時代に「業務システムを作る」プレイヤーは大きく3つに分類できる。PortXはどれとも正面衝突しない。

プレイヤー A — 確率的(Probabilistic)

Vibe Coding

Cursor / Copilot / Devin / Claude Code

強み：個人開発者・スタートアップの内製を爆速にする。デモまでは速い。

届かない領域：仕様書が無いため**"動くこともある"**確率的な振る舞い。エッジケースで壊れ、変更が予期せぬバグを生む。**エンタープライズの本番稼働・運用保守には構造的に到達できない**。

プレイヤー B

UI/GUI 生成ツール

V0 / Bolt / Lovable

強み：誰でも画面を作れる。プロトタイプ・MVP・社内ツール・LPに強い。

届かない領域：仕様書(SSOT)がないため**プロトタイプ止まりで本番稼働できない**。業務例外・整合性・変更影響追跡・運用保守が構造的に不可。

プレイヤー C

従来型エンタープライズSI

富士通 / NEC / NTTデータ / 大手SIer

強み：大規模システムの構築実績と顧客信頼。

届かない領域：人月で稼ぐ構造のため、自社で効率化できない。経産省が「低位安定の共依存」と命名した構造そのもの。

PortX — 決定論的 (Deterministic)

Formula × FDE

Spec-Driven Development

PortXは A・B・Cのどれとも正面衝突しない。投資しているのは エンタープライズ業務SIに特化したFormula Platform Layer (Spec-Driven Development 基盤) と、顧客現場で業務を仕様書に変換するFDE組織。
この2つが密結合しているため、プラットフォームだけ模倣されても複製できない。大企業の中核業務を、毎回正しく動く本番品質で実装できる唯一の構造。

ビジネスモデル

3階層の収益モデル

案件ごとに売り切るのではなく、**プラットフォーム上で継続収益を積み上げる**構造。

フロー収益
構築支援費用

最初に立ち上げる・追加で作る時の費用。要件整理・設計・実装・テスト・UAT支援まで。

課金：人月単価（標準工数ベース）

ストック収益（主力）
運用基盤費用

本番運用・保守の月額費用。認証・監視・バックアップ・問い合わせ対応・変更影響分析まで。

課金：月額固定（Standard / Pro / Enterpriseの3ティア）

オプション
高度運用支援

ハイパーケア・月次レビュー・優先対応など、追加の人手支援。

課金：月額オプション（必要な顧客のみ）

コスト構造の特徴：AIで効率化された分を値引きとして還元しない。**効率化の果実は会社に蓄積し、プラットフォーム強化に再投資**する。案件ごとに消費されるのではなく、プラットフォーム上でアプリが稼働し続ける構造。

市場規模 ① — 売上規模帯ごとに見る、エンタープライズSI市場

同じ「SI市場」も、企業の売上規模ごとに競合構造が違う

Formulaの「入り方」は従来SIと違うので、規模帯ごとに勝ち筋・競合・取得可能シェアが変わる。**PortXは規模帯を選んで攻める。**

売上規模帯	国内対象社数	主な競合 / Formulaの勝ち筋	PortX位置づけ	取得目標
A. 1兆円超	約40社	大手SIer + 内製AI組織。narrow wedgeで一部門を取り、横展開する経営直轄型。	**挑戦市場**	**5–10%**
B. 5,000億〜1兆円	**約80社**	大手SIer + 業界特化SIer。**ユーザー部門起点**で「競争優位の実装」を取りに行く。	**主戦場 ①**	**10–15%**
C. 1,000億〜5,000億	**約700社**	中堅SIer + パッケージSaaS。**「フルスクラッチのようなSaaS」**として、SIかSaaSかの二択を超える。	**主戦場 ②**	**10–15%**
D. 500億〜1,000億	約1,500社	パッケージSaaS + 内製ローコード。**従来SIには経済性が成立しない領域**。Formulaで初めて取れる。	**新規開拓市場**	**10–15%**
E. 500億未満	多数	汎用SaaS。narrow wedgeの経済性が出ないため対象外。	対象外	—

※ 業種は製造業+物流業に絞った社数。取得目標は新規開発+運用保守の合算ベース。**主戦場B+Cで先行確立し、AとDに段階的に展開する**のが基本シナリオ。

PortX

市場規模 ② — 主戦場B+Cの定量化

主戦場(B+C 約780社)だけで、年間SI支出は1.2〜1.5兆円

製造業+物流業 × 売上1,000億〜1兆円 × 中核業務 (ERP/SCM/生産/WMS/TMS/調達/品質) のSI支出を積み上げた数字。

主戦場 ① — 売上 5,000億〜1兆円

大手SIerの牙城を、ユーザー部門起点で崩す

対象社数	中核業務SI支出/社・年
約80社	約50〜80億円

市場サイズ

約4,000〜6,000億円/年

取得目標 10–15% → **PortXのTAM=400〜900億円**

主戦場 ② — 売上 1,000億〜5,000億円

「SIかSaaSか」の二択を超える層

対象社数	中核業務SI支出/社・年
約700社	約10〜15億円

市場サイズ

約7,000億〜1兆円/年

取得目標 10–15% → PortXのTAM=700億〜1,500億円

B+C 合計　主戦場の年間SI支出　**1.1〜1.6兆円**

取得目標10–15%

→ PortXのTAM 約1,100〜2,400億円/年

※ 中核業務SI支出/社・年は、JEITA/IDC/総務省のIT支出データから売上規模帯別に按分推計。新規開発+運用保守の合算ベース。詳細はAppendix参照。

PortX

市場規模 ③ — 主戦場で勝った後の展開シナリオ

主戦場B+Cで先行確立すると、A・D・隣接業種に広がる

Formulaという「サービス」は、土台を早く作り始めて磨くほど後から追えなくなる。**B+Cで実績を積むことが、AとDへの最短ルート。**

PHASE 1 — NOW (2024〜2027)

主戦場 B+C で先行確立

製造業・物流業の中核業務(SCM/生産/WMS/TMS等)に narrow wedge で深く入る。Formulaの土台を磨き、再利用可能な業務テンプレートとSkillを蓄積。

→ PortX TAM 約1,100〜2,400億円

PHASE 2 — NEXT (2027〜2030)

A帯(1兆円超)とD帯(500〜1,000億)への上下展開

A帯：B+Cで蓄積したテンプレートを武器に、大手SIerの牙城に部門単位で食い込む。
D帯：Formulaで初めて経済性が成立する層。SaaS的提供で一気に取りに行く。

→ 追加TAM 約1,500〜3,000億円

PHASE 3 — VISION (2030〜)

業種横展開 (小売・金融・医療)

中核業務SIの構造課題は製造業以外にも共通する。業務テンプレート群が成熟すれば、業種をまたぐ展開コストが急減する。

→ 追加TAM 数千億円〜

VISION (長期)

Formula自体を他SIerに提供

Formula Platform Layerを業界共通基盤として開放。PortXは原盤ホルダーとして、業界全体のSI生産性を引き上げるレイヤーになる。

→ TAM = 国内SI市場全体の構造変革

先行優位のロジック：Formulaは "ツール" ではなく "サービス" のため、土台を早く作り始めて磨いた者ほど強い。フロンティアモデルが進化するたびに **既存の業務テンプレート・Delta履歴・整合性ルールが自動的に強化される。**後発がゼロから同じものを積むには、PortXが先に過ごした年数が必要になる。

PortX

ターゲット顧客と勝ち筋

Narrow Entry × Broad Vision

主戦場B+C(売上1,000億〜1兆円の製造業・物流業)に、ユーザー部門起点で深く入る。

ターゲット顧客
製造業・物流業のユーザー部門

売上 **1,000億〜1兆円** の中堅〜大手企業が中心。情シス経由ではなく、**業務で本当に困っている現場部門**に直接アプローチ。

在庫・納期・受注配車・生産調整・調達・品質など、**企業固有性が高く、部門横断の中核業務**に狙いを定める。

勝ち筋
5つのコア戦略

① ユーザー直結（情シス経由しない）
② ミッション起点（機能でなく業務変化で語る）
③ 早期具体化（動くToBeを商談段階で提示）
④ 一気通貫（要求整理から運用まで分断しない）
⑤ 複利化（案件ごとに会社資産が積み上がる）

勝ち筋の本質：狭く深く入ることで、大手SIerともパッケージSaaSとも正面衝突しない。**部門単位の実績がFormulaの業務テンプレートとして蓄積され、横展開のたびにコストが下がる複利構造。**

ロードマップ

PortXとFormulaの進化

プラットフォームと組織能力を同時に積み上げていく。

PHASE 1 — NOW (2024〜2027)

主戦場B+Cで先行確立

- 売上1,000億〜1兆円の製造業で実績
- Formulaパイプラインの完成・運用
- FDE組織の立ち上げ・スケール
- 業務テンプレート資産の蓄積

PHASE 2 — NEXT (2027〜2030)

A帯・D帯への上下展開

- 売上1兆円超(A)に部門単位で食い込む
- 売上500〜1,000億(D)にSaaS的提供
- 物流・隣接業種への展開
- 運用基盤ARRの積み上げ

PHASE 3 — VISION (2030〜)

エンタープライズ業務ソフトウェアの標準

- Formulaを他のSIerにもライセンス提供
- 業界標準プラットフォーム化
- 小売・金融・医療への業種拡大
- エンタープライズSIの構造を変える

核となる戦略：PortX自身がFormulaを使い倒してデリバリー能力を高め、その実績と知見を武器にプラットフォーム自体を進化させる。**自社利用と外部提供の両輪**で成長する。

会社概要

会社名	株式会社PortX / PortX Inc.
代表者	石田 寛成
資本金	1億円（累計調達金額7.2億円）
設立	2019年12月6日
本社	東京都新宿区新宿2-5-12 FORECAST新宿AVENUE 6F
事業	エンタープライズ業務ソフトウェア「Formula Platform」の開発・運用

Appendix

市場規模の算出仮説と前提

本資料の「PortX中核業務SI市場」の数字の根拠と算出過程。

市場定義

PortX中核業務SI市場 = 売上500億円以上の製造業+物流業企業における、調達・購買・生産・販売・受注・物流・在庫・品質管理というバリューチェーン中核業務を支えるITシステムの新規構築・運用保守にかかる年間支出。

新規開発市場 ≒ 約4,000億円/年 の算出式

国内プロジェクトベース市場 (IDC 2024) × **製造業+物流業シェア** × **中核業務比率** × **大手企業シェア**

= 3.6**兆円** × 35% × 45% × 70%

≒ 約3,970億円/年 → 約4,000億円/年（上限レンジ採用）

運用保守市場 ≒ 約1.0兆円/年 の算出式と仮説

国内マネージドサービス市場 (IDC 2024: 4.14兆円) × **製造業+物流業シェア** × **基幹系業務比率** × **大手企業シェア**

= 4.14**兆円** × 35% × 45% × 70%

≒ 約4,560億円/年（狭義: IDCマネージドサービス経由の推計）

「1兆円」の仮説：IDC統計の狭義マネージドサービスに加え、JUAS調査の「IT予算の8割が保守運用」という構造を踏まえ、**社内IT人件費・SIer保守契約・パッケージ保守費などのマネージド外保守を含める**と、狭義推計の約2〜2.5倍に膨らむ。PortXの運用基盤費用は「カスタム開発の保守＋クラウドマネージド」を両方置き換えるため、両者を合算した**約1兆円/年が現実に置換可能な上限**と仮説を立てた。この数字は「8:2の構造歪み」が解消される前提で成立する。

2030年予測の前提

- ベースライン成長: CAGR +8%（IDC Japanのプロジェクトベース市場CAGR 9.1% / マネージドサービス 5.1% の加重平均）
- 伸びしろシナリオ×2.5: 日本の製造業IT投資比率 1.0% → 国際水準3.3%への戻し（NRI/Gartner）
- PortXが勝ち筋として狙うのは、ベースラインではなく**伸びしろ解消を主導するポジション**

Appendix

主要出典 — 全て公的機関・一次情報

本資料の数字・主張は以下の一次情報を根拠としている。

政府機関・公的統計

① 経産省 DXレポート（2018）
「2025年の崖・12兆円/年」の原典。

② 経産省 DXレポート2.1（2021）
「ユーザー企業とベンダー企業の低位安定の共依存」公式表現。

③ 経産省 レガシーモダン化委員会総括レポート（2025.5）
レガシー残存61%・大企業74%・IT予算9割保守の追認。

④ 経産省 IT人材需給調査（2019）
2030年IT人材不足最大79万人の原典。

⑤ 総務省 平成30年版 情報通信白書
IT人材72%ベンダー所属 vs 米国65%ユーザー所属の原典。

⑥ JUAS 企業IT動向調査 2025
IT予算の新規開発：保守運用比率（2:8）、業種別IT投資。

市場調査会社（民間）

⑦ IDC Japan 国内ITサービス市場予測（2024-2030）
国内IT市場27.9兆円、ITサービス7.0兆円、SI 3.6兆円、マネージドサービス4.1兆円。

⑧ 矢野経済研究所 国内企業IT投資調査（2025）
国内民間IT市場15.8〜16.7兆円。

⑨ 富士キメラ総研 業種別IT投資（2024）
製造業+物流運輸のIT投資シェア、DX投資26.0%比率。

⑩ 矢野経済研究所 ERP/工場DX市場（2025）
ERP市場1,684億円、工場デジタル化1.84兆円、PLM 761億円。

⑪ 富士経済 次世代物流システム・サービス市場（2024）
次世代物流市場7,542億円→2030年1兆1,670億円。

⑫ NRI / Gartner IT投資国際比較
日本製造業1.0% vs 米国3.3% vs 欧州2.6% の原典。

数字は2025-2026年調査時点の暫定値。最新版・詳細計算はお問い合わせください。