

ホワイトペーパー：

物流システム刷新を「後発の強み」に変える現場ファースト DX

～「現状維持」「標準化」の罠を越え、10 年先も戦える物流基盤をつくる実践アプローチ～

発行日: 2026 年 9 月 発行元: シーオス株式会社

この資料でわかること：

- ・物流システム刷新が進まない本当の理由と、レガシーシステムを放置する経営リスク
- ・「現行踏襲」と「パッケージ標準化」が現場の生産性を下げる構造
- ・AI・モックアップ・段階移行を活用し、現場定着までつなげる 4 つの実践ステップ
- ・後発企業だからこそ、物流 DX で先行企業を追い越せる可能性と、そのための推進体制

このような方におすすめ：

- ・老朽化した基幹システム、WMS、周辺システムの刷新を検討している物流・SCM 責任者
- ・AS400 やオフコン、Excel、手作業が残る現場を抱える情報システム・DX 推進部門の方
- ・多拠点で異なる業務ルールを整理し、物流システムの標準化と競争力向上を両立したい方
- ・システムリプレイスの失敗や現場定着への不安から、刷新プロジェクトを進めきれずにいる経営層・プロジェクト責任者

はじめに：システム刷新は「古い仕組みを新しくする仕事」ではない

「動いてはいるんです。ただ、作った人はもう誰もいません」。基幹システム刷新のご相談をいただいた企業で、情報システム部門の責任者が口にした言葉です。二十年以上前に導入されたオフコンが、いまでも日々の受発注と在庫を支えている。仕様書は残っておらず、改修を頼めるのは特定ベンダーの担当者ただ一人。それでも止まらずに動いている以上、今年も触らないという判断が更新されてきました。

しかし実際には、人手不足、EC 化、物流を取り巻く制度・環境の変化、AI をはじめとする技術進化によって、ビジネスの前提条件そのものが変わり続けています。

問題は、システムが古いこと自体ではありません。新しい顧客要件にシステムが対応できず、現場が Excel や手書き伝票で補完する。拠点ごとにローカルルールが増え、担当者の経験で何とか回す。軽微な改修でも影響範囲が分からず、高い費用と長い期間が必要になる。こうした状態が積み重なると、システムは業務を支える基盤ではなく、事業成長の「足かせ」になります。

一方で、刷新を先送りしてきた企業には大きな機会もあります。先行企業が経験してきた失敗や、パッケージに業務を合わせることで生じた現場負荷を教訓にしながら、AI などの新しい技術を活用し、初めから「変化を前提にした仕組み」を設計できるからです。本資料では、物流システム刷新を単なるリプレイスではなく、競争優位をつくる物流 DX へと変えるための考え方と進め方を整理します。

第1章：「動いているから触らない」が経営リスクに変わるとき

ブラックボックス化は、障害よりも静かに事業を蝕む

長年利用しているシステムほど、「詳しい担当者が退職・異動している」「仕様書が更新されていない」「改修の経緯を知る人がいない」といった状態になりがちです。日常業務が動いている間は問題が見えませんが、障害時に復旧方法が分からない、業務変更に伴う改修ができない、調査だけで費用と期間が膨らむ、といったリスクが一気に顕在化します。

さらに深刻なのは、障害が起きなくても機会損失が生じることです。たとえば新しい荷主から、これまでとは異なる出荷条件や在庫管理粒度、データ連携を求められたとします。本来であればビジネス側が「受けるかどうか」を判断すべきところ、「今のシステムでは対応できない」という理由で案件を諦める。これはITの問題ではなく、経営の自由度をシステムに奪われている状態です。

Excel・手作業の増加は「現場の工夫」ではなく、システムとのズレのサイン

既存システムで処理できない業務が増えると、現場はExcel、CSV、紙、口頭確認などで隙間を埋めます。短期的には、現場の工夫によって業務が止まらずに済みます。しかし、この補完運用が常態化すると、同じデータを複数のファイルへ転記する、担当者ごとに管理表が異なる、集計のたびに数字を突き合わせる、といった「見えないコスト」が積み上がります。

この状態で「今と同じものを新しい技術で作り直す」だけでは、課題は解消されません。むしろ、長年積み上がった非効率まで新システムへ移植してしまうことになります。物流システム刷新の出発点は、既存機能の棚卸しではなく、「現在の業務のどこに価値があり、どこが過去のシステム制約によって生まれた作業なのか」を見極めることです。

第2章：物流システム刷新を失敗させる「2つの罠」

2-1 罠① 現行踏襲——技術的負債を最新環境へ運ぶだけになる

刷新プロジェクトで最も安心感があるように見えるのが、「今と同じ機能をそのまま新しくする」という現行踏襲です。業務変更が少なく、現場への説明もしやすいため、一見するとリスクが低いように思えます。しかし、既存システムの制約を補うために作られたExcelや例外運用まで含めて現状を再現すると、課題の構造は何も変わりません。

たとえば、基幹システムが閉鎖的でWMS、TMS、BIなどへ必要なデータが連携できず、各システムが個別にデータを保持しているケースでは、部門ごとの部分最適が進みます。その間をExcelの「バケツリレー」でつなぐ運用が続けば、システムを刷新してもDX投資の効果は限定的です。必要なのは、既存機能を正確にコピーすることではなく、データと業務の流れを改めて設計することです。

2-2 罣②「標準だから安心」とパッケージに現場を合わせる

もう一つの罣が、「パッケージの標準機能に業務を合わせれば、コストもリスクも抑えられる」という考え方です。もちろん、標準化できる業務まで個別化する必要はありません。しかし物流現場には、商材特性、荷姿、設備、顧客別サービス、拠点制約など、物理世界に起因する例外が数多く存在します。それらを一律に切り捨てると、システム上は標準化できても、現場側に手作業が戻ります。

分かりやすい例が画面です。業務で使わない入力欄や選択肢が大量に表示されると、作業者は毎回「ここは入力するのか」「どれを選ぶのか」を判断します。1回は数秒でも、毎日何百、何千回と操作される現場では、生産性や入力品質、教育コストに影響します。必要な情報だけが自然な順序で表示され、マニュアルを熟読しなくても操作できる。こうした使いやすさは、現場定着を左右します。

たとえばシステム刷新の内製化を決断した N 社の事例では、「IT のためにビジネスを我慢しない」という考え方を示しています。自社の業務要件にシステムを合わせ、必要な機能を継続的に変えられる状態をつくるのが、物流や小売の強みを守ることにつながるという示唆です。

第 3 章：後発企業が DX リーダーへ躍進する「現場ファースト」の設計

3-1 必要なのは「完成品」ではなく、変化に追従できる可変力

10 年先まで同じ業務・顧客要件が続くことを前提に、全機能を固定して作ることは現実的ではありません。重要なのは、変化に合わせて機能を足し引きできる「可変力」です。コア機能を先に実装し、周辺機能を段階的に追加する。外部環境や顧客要件が変われば必要な機能を追加する。こうした設計は、刷新を一度きりの投資ではなく、継続的に進化する事業基盤へ変えます。

3-2 AI の役割は「判断」ではなく、複雑な現場情報を構造化すること

AI 時代のシステム刷新では、これまでコストや期間の制約で拾いきれなかった現場の暗黙知や例外処理を、設計へ組み込みやすくなります。大量の画面、帳票、業務フロー、ヒアリングメモなどを AI で横断解析し、画面・帳票・項目・機能の一覧へ整理する。業務要件からモックアップ画面を素早く作る。こうした活用により、人は資料整理ではなく「何を残し、何を変えるか」という判断に集中できます。

ただし、AI は「その例外が自社の強みなのか、過去の習慣なのか」を判断できません。たとえばベテラン作業者の判断した定性的な理由を、条件としてデータ項目へ構造化することは AI を活用することで効率化できます。しかし、その条件を残すべきか、別の業務設計で解消できるかは人が判断します。AI を使うほど、「問いを設計する力」と不要業務を捨てる判断力が重要になります。

3-3 分厚い仕様書より「動く画面」で現場と合意する

要件定義で起きやすい問題は、IT に不慣れな現場へ分厚い仕様書を渡し、「これで問題あり

ませんか」と確認してしまうことです。文章だけでは実際の操作を想像しにくく、稼働直前になって「思っていた画面と違う」「この項目がない」といった認識ずれが発覚します。そこで有効なのが、早い段階からモックアップを用意し、実際の画面を見ながら議論する方法です。「この項目は必須か」「検索条件を追加したい」「表示順を変えたい」といった具体的な意見が出やすくなり、要件の抜け漏れや手戻りを抑えられます。たとえばシステム刷新プロジェクトを現場優先の思想で進めた H 社の取り組みは、文章の仕様書だけに頼らず、現場が画面イメージを見ながら要件を具体化することの有効性を示すものです。

第 4 章：失敗リスクを抑え、現場定着までつなぐ 4 つの実践ステップ

4-1 Step1 業務の可視化・棚卸——「なぜ、その手順が必要か」を問い直す

最初に行うべきは、プログラムの機能一覧ではなく、実際の現場ワークフローの可視化です。受注、在庫確認、出荷、請求などについて、誰が、何を受け取り、どう処理し、どこへ渡すかを整理します。そのうえで各手順に「なぜ必要なのか」を問いかけます。顧客・品質・事業価値に直結する業務は残す。単なる習慣、重複、待ち時間、旧システムの制約に合わせるだけの作業は捨てる。価値はあるがやり方が悪いものは再設計する。この仕分けがシステム要件の質を決めます。

4-2 Step2 業務の統合・標準化——例外を消すのではなく、意味を見極める

複数拠点の物流ネットワークを有する企業では、「A 倉庫ではこの帳票」「B 倉庫では別の締め時間」「C 倉庫だけ独自コード」といったローカルルールが増えています。これらを整理せずシステム開発へ入ると、拠点ごとの差分がそのまま要件となり、システムが複雑化します。まず業務プロセスを統合し、誰でも回せる標準業務をつくることが重要です。ただし、標準化はすべてを同じにすることではありません。商材特性や顧客サービスなど、事業上の意味がある差分は残す必要があります。ここで求められるのが「設計者」の役割です。AI などで整理された例外を見ながら、残すべき強みと、単なるローカル習慣を峻別し、システム要件へ翻訳します。

4-3 Step3 移行計画——ビッグバンを避け、小さく切り替える

物流や基幹業務は「止められない」ため、一括切替への恐怖が刷新を止める大きな要因になります。そこで、全面リプレイスありきではなく、機能ごとの段階切替や、新旧システムが同じデータを見ながら並行稼働する方式など、複数の移行シナリオを検討します。問題が起きた場合に対象機能だけ切り戻せるようにすれば、影響範囲を限定しながら前進できます。M&A や事業再編の背景を持つ L 社では、複雑化した基幹システムを一度に切り替えるのではなく、新旧並行稼働と段階移行によって停止リスクを抑える考え方を示しています。「完璧な切替計画を一度で作る」より、「小さく切り替え、安全を確認しながら進める」ことが、止められない業務の刷新では現実的です。

4-4 Step4 実装・定着——経営・現場・IT をつなぐ推進体制をつくる

最後の成否を左右するのは、システムの機能よりも推進体制です。経営は投資目的と KPI を示し、現場は実態と例外を伝え、IT は技術的な制約と実現方法を提示します。しかし、それぞれの言葉はそのままでは噛み合いません。経営の狙いを現場が動けるテーマへ変換し、現場の暗黙知をシステム要件へ翻訳し、技術上の制約を経営が判断できる言葉に戻す「翻訳者」が必要です。

また、通常業務を抱えながら大規模な刷新を進める企業では、業務棚卸、部門間調整、要件化、移行計画、PMO といったリソースが不足しがちです。外部支援を活用する場合も、プロジェクトを丸投げするのではなく、投資目的、残す業務・変える業務、優先順位などの主導権は自社に残し、不足する推進力・翻訳力・設計力を補完するという考え方が重要です。

結びに代えて：レガシー脱却を「守りの IT 投資」で終わらせない

物流システム刷新は、老朽化したシステムを新しくするための修繕ではありません。ブロックボックス化、サイロ化、Excel 依存といった過去の制約から脱却し、事業の変化へ素早く対応できる基盤をつくるための戦略投資です。現状維持を続ければ、システムが原因で新しい要件に応えられず、現場負荷と機会損失が静かに増えていきます。

一方、刷新が後発になった企業は、先行企業の失敗を学び、AI による現状整理やモックアップ、段階移行といった新しい手段を最初から取り入れられます。大切なのは、パッケージかオーダーメイドかという製品選択から議論を始めるのではなく、「自社が守るべき強みは何か」「10 年先にどのような変化へ対応したいか」を起点にすることです。

システムのために現場を歪めるのではなく、現場とビジネスの強みをシステムで支える。必要な業務は残し、不要な業務は捨て、変化に合わせて機能を進化させる。その設計ができれば、刷新を先送りしてきた企業にも、物流をコストセンターから競争優位の源泉へ変える機会があります。まずは現状の業務とシステムがどこでズレているのかを可視化するところから、次の 10 年に向けた物流 DX を始めてみてはいかがでしょうか。

シーオス株式会社 について

シーオスは、ロジスティクス分野に 25 年以上特化してきたソリューションカンパニーです。

「ロジスティクス」を単なる物の移動ではなく、企業価値に直結する重要領域と捉え事業を展開してまいりました。創業以来、ロジスティクス領域に特化した専門チームを擁し、現場理解と戦略立案の双方に強みを持つパートナーとして、製造・小売・EC など数多くの企業様の課題解決に伴走しております。

戦略コンサルティングからシステム開発、WMS（倉庫管理システム）や TMS（輸配送管理システム）、最先端の AMR（自律型搬送ロボット）ソリューション、さらには現場業務の受託に至るまで、フルラインアップのサービスを提供しています。