

# 短期型実装を支えるITサービスマネジメント

## 1. ビジネス環境の変化への迅速な対応

近年 IT サービスは、ビジネスでの価値創出を実現する上で必要不可欠な存在となっており、ビジネス環境の急速な変化の中では、IT サービスもビジネス要求へのより迅速な対応を迫られています。新たな IT サービスの提供や変更期間短縮のため、それを支える IT システム構築の迅速化が求められており、その取り組みとして、以下のトレンドが挙げられます。

- IT 基盤構築期間の短縮：IT 基盤としてのクラウド・コンピューティングの活用
- 開発期間の短縮：アジャイルなど新たなアプリケーション開発手法の活用
- 開発から運用への引き継ぎの短縮：開発部門と運用部門の密接な連携といった新たな試み

これらの新たなトレンドを、従来型の情報システム部門の構築・運用の形態および手法に取り入れる企業も少しずつ増えています [1]。これらのトレンドは、競合他社への優位性を保つために、サービスを迅速に市場に投入したいというビジネスや開発側からの要件と、その一方で頻繁な更改でサービスの中断や品質低下を招かないようにしたいという運用側の要件を一致させようとする試みです。当コラムでは、これらのトレンドの中から「開発から運用への引き継ぎの短縮」を取り上げ、IT サービスマネジメント（以下、ITSM）の視点から、どのようなアプローチが検討できるかを考察します。

## 2. 従来型開発と短期型実装に求められる運用とは

従来の典型的な開発・実装手法（以下、従来型開発）は、数カ月から年単位の期間を必要とするウォーターフォール型の開発です。それに対して、昨今注目されている頻繁な要件変更に対応できる開発・実装手法（以下、短期型実装）の典型は、クラウドの利用による迅速な開発・テスト・本番環境構築であり、アジャイル型やスパイラル型の開発アプローチです。短期型実装の場合、短期間かつ頻繁に開発が繰り返されることから本番への移行を迅速に行う必要があり、従来型開発とは求められる運用に相違があります。ここでは開発・実装手法の特徴に応じた運用要件を整理します。

それぞれの開発・実装手法における大きな差異は、要件定義から本番移行までの期間の長短です。短期型実装では短期間での本番移行が求められ、その頻度も月単位／週単位／日単位や、極端な場合には時間単位での更新が要求されます。同時にリスクへの対応も必要です。本番移行の短期間化や頻繁化に対応するためには、毎回の変更内容の単純化や、変更範囲の限定などにより、リスクを低減させることが大切です。

短期型実装の特徴と、それに対応するための運用要件を表1に示します。

これまで提示したように、短期型実装は従来型開発とは異なり、小さいトランジション（開発環境から本番環境へのサービス移行）が定期的が発生することになります。この定期的なトランジションを円滑に実施するためには、開発と運用とのコミュニケーションや連携、フィードバックなどが不可欠であり、開発と運用との間でこれまで以上に密接な関係が求められます。

業務の特徴や要件を考慮し、従来型開発が適しているのか、短期型実装が適しているのか見極め、さらに短期型実装でも毎時間など超短期の本番移行が必要なかを判断し、適切な開発スタイルと運用との連携を選択する必要があります。次章では、短期型実装の特徴を前提に、短期型実装を適用する運用に対して、ITSM がどのように貢献できるのか、その位置付けについて整理します。

表1. 短期型実装の特徴と運用要件

短期型実装の特徴	短期型実装に対応するための運用要件（太字）
実装までの期間： 短い	●開始から本番稼働までが数週間や数日といった短期間での実装となるため、 <b>変更要求への対応処理を迅速化する。</b>
変更の頻度： 多い	●毎月、毎週、毎日、毎時など頻繁な変更要求が発生するため、 <b>変更の対応処理の迅速化に加えて変更の実装サイクルを短縮する。</b>
変更の規模： 小さい	●変更内容は個別要求への対応であり、変更範囲／変更内容は小規模であるため、 <b>個別の小規模変更に適した変更処理を確立する。</b>
変更のリスク： 小さい	●個々の変更のリスクは小さいが、一連の複数の変更における全体リスクは必ずしも小さくないため、 <b>個々の変更リスクと全体リスクの両面でリスク管理を実施する。</b>
サービス展開の タイミング： 定期的	●サービスの展開が短期間で頻繁に発生し、定期的となるため、 <b>組織体制や役割／プロセスなどが定期的な変更要求・実装に対応できるようにする。</b>
利用者からの フィードバック： 定期的	●利用者からの定期的なフィードバックが新たなサービス要件となるため、 <b>サービスとその運用の両面で、サービス要件の受け入れと反映を適切なサイクルで処理する。</b>
開発の容易性： 容易	●小規模で、環境構築や結果確認、利用部門による検証が容易であり、並行した複数の開発実装となるため、 <b>運用面での全体のガバナンスや整合性を確保する。</b>
開発の継続性： 必要	●要件の変化に応じたサービス拡張など定期的で継続した開発を行うため、 <b>同じサービスへの継続的な変更要求に対して、迅速で効率的、確実な対応を行う。</b>
開発・運用の連携： 緊密	●開発から本番への展開が定期的であるため、 <b>開発局面と運用局面を一体化し、開発と運用の“縦割り”を排除して、緊密な連携を行う。</b>

### 3. ITSMのライフサイクルにおける短期型実装の位置付け

ITIL (IT Infrastructure Library) は、ITシステムの運用にフォーカスしたベスト・プラクティスと思われがちですが、実はITSMのライフサイクルの全フェーズを網羅しており、運用と開発の関係についての記述も数多く含まれています。開発と運用の連携に関しては、「サービストランジョン」[2]の中での「変更管理」「リリース管理および展開管理」のプロセスが特に深く関係しています。

ITILでは、変更が原因で発生するサービス品質の低下を最小限にするために、ITシステムの変更は慎重に取り扱うことが重要であると考えます。変更管理においては、変更を影響度(ビジネス影響、変更リスク、変更コスト)に基づいて以下の3つのタイプに分類し、それに見合った変更手順を順守することと定めています。

- 標準的な変更 (Standard Change) : 事前承認済みの変更
- 通常の変更 (Normal Change) : 事前承認処理 (必要に応じて変更諮問委員会) が必要な変更
- 緊急の変更 (Emergency Change) : 緊急の承認会議 (緊急変更諮問委員会) が必須の変更

リリース管理においては、変更を技術的観点に基づいて慎重にスケジュール化(類似変更の取りまとめ、大規模変更の分割)して、変更がうまくいかなかった時の切り戻し(ロールバック)の手順も必ず準備するように定めています。

以前からITILでは、リリースを集約して頻度を減らすことでリスクを抑制するという考え方が定着しています。しかしさまざまなテクノロジーにより、従来多くの労力が必要であったリリース・ユニット構築やロールバック用の構成ベースライン作成が、最小限の時間と労力で実現できるようになっています。変更管理とリリース管理にこれらの技術を適用することで、短期型実装への対応を促進することができます。次章では、そのアプローチについてご紹介します。

### 4. 変更/リリース管理における短期型実装への対応

短期型実装の変更プロセスには、ITILの提唱する「標準的な変更」の適用が不可欠です。「標準的な変更」とは、パスワードのリセットなど、リスクが低く変更範囲が限定的で毎回一定の手順で適用できる変更を事前に承認し、素早く適用するアプローチです。短期型実装の開発部門と運用部門が協議し事前承認を得た上で、日々の運用に影響がない変更は、「標準的な変更」と定義することができます。また、前述のさまざまなテクノロジーを活用し、手作業の変更適用手順を自動化できれば、従来は一連の承認プロセスが必要であった「通常の変更」を、「標準的な変更」に移行できる可能性が出てきます。図1はIBM Tivoli Unified Process (以下、ITUP) ツール [3]における変更要求(Request for Change: 以下、RFC)の処理フローです。「通常の変更」を太枠内の「標準的な変更」に順次移行することで、変更の成功率向上や迅速なビジネス要件への対応を促進する効果があります。

さらに、短期型実装を支える以下のようなテクノロジーにより、展開(配布)・リリース(本番適用)の自動化が可能になります。

- 多数の管理対象に対するソフトウェア自動配布

- 機能、回帰、負荷、統合などのテスト仮想化/自動化
- サービス・カタログを通じてITリソースを自動的に割り当てるプロビジョニング
- スナップショットやキャプチャー機能を利用した適用失敗時のロールバック

ITSMでは、展開内容の配布と統合処理、統合テストと検証、展開パッケージの作成と展開の手順を定義しています。従来型開発ではこれらの手順を展開ごとに局面化して実施しますが、短期型実装では、パッケージ展開を類型化し反復する特性があるので、図2の太枠内の手順を標準化できます。展開手順を標準化・自動化することにより、配布と本番適用のための時間と労力を大幅に削減し、その成功率と効率向上に大きく貢献します。

### 5. ITSMを短期型実装に対応させる上での考慮点

これらの短期型実装を支えるテクノロジーを、従来型開発を対象としたITSMのプロセス・ワークフローへ実際に適用する際には、以下の通り3つの大きな課題が想定されます。

#### (1) ガバナンス

「すべての変更は変更管理プロセスを通じて実施する」。これはITSMの変更管理における大原則ですが、短期型実装環境で展開・リリースの自動化を促進することと引き換えに、変更管理プロセスをバイパスしてしまうリスクが高くなります。

従来型開発と短期型実装とで変更実施担当組織が異なる場合でも、全社共通の変更管理プロセスを順守するように共通認識を持つ必要があります。

#### (2) 役割と責任

例えば従来型開発でのテストは「サービスの妥当性確認およびテスト」のITSMプロセスが実施しますが、短期型実装では開発部門が実施する場合があります。このように両者の役割と責任に違いがあることで起き得る以下のような課題に対応する必要があります。

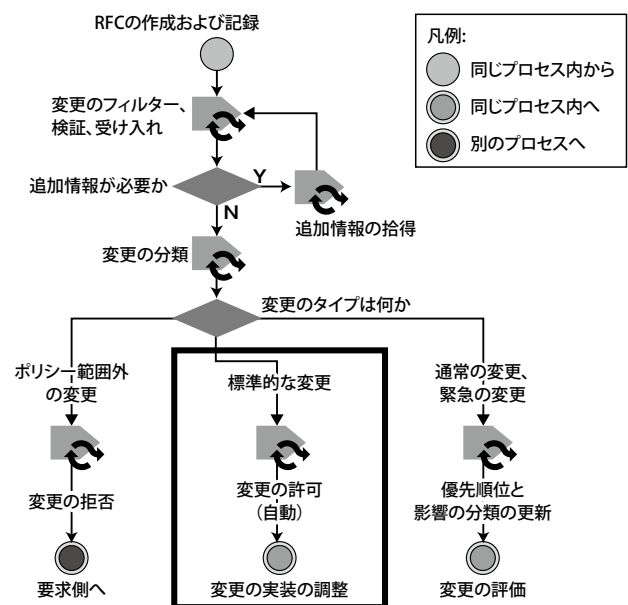


図1. 標準的な変更の適用

- ・開発と運用で、役割と責任の認識が一致しない
- ・変更による影響範囲の理解に隔りがある
- ・変更による影響の指標として、従来型開発はCI(Configuration Item:構成アイテム)数、短期型実装はユーザー数など、各責任部門が異なる指標を使うと、影響を正確に判断できない

### (3) 運用業務への影響

まず変更管理では、短期型実装における「標準的な変更」の承認基準や運用業務への影響の有無を判断するための基準を確立する必要があります。さらに基準確立と併せて、「標準的な変更」以外の変更の影響評価に、短期型実装の開発関係者の参画を促進することも重要です。

また、リリース管理および展開管理では、ロールバック時の影響範囲と各復旧プラン所要時間について、短期型実装部門と運用部門が連携して迅速に予測できるようにして、リリース失敗時のロールバック基準を新たに確立する必要があります。さらに成功、失敗にかかわらずリリース実行時のエビデンスと証跡を取得するために、自動化・標準化された手順を開発する必要があります。

## 6. 今後の展望

既存のITSMプロセスは、自社開発された「歴史のある」サービスや、ミッション・クリティカルなサービス提供を支えている従来型開発と整合性の取れたものでした。本稿では、短期型実装によるサービス提供も支援できるようにITSMプロセスを改善するアプローチを紹介しました。

短期型実装の実績が増える中、サービスを提供するインフラとその上で稼働するアプリケーション、さらにその運用を、あらかじめ短期型実装に最適化して1つのサービスとして提供するモデルも登場しています。このモデルを利用したサービス提供が、従来型

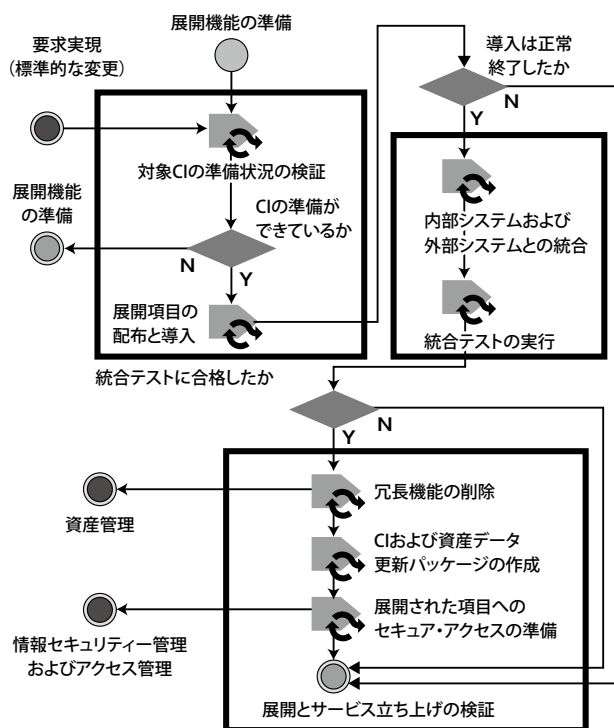


図2. パッケージ展開における標準化・自動化対象の手順

開発と既存のITSMから独立して実施できるのであれば、当面これを既存のITSMと並存させて実施する選択肢もあります。あるいはこれらのモデルが成功し、実績を積むことができれば、従来型開発と既存のITSMを新しい提供モデルへ統合するアプローチも考えられます。しかしいずれの場合も、プロセスの標準化と改善が不可欠であり、本稿のアプローチは有効です。テクノロジーを適切に活用し、人が管理プロセスの各種活動を実施することで、価値あるサービスを顧客に提供するというサービスマネジメントの原則は不変です。



宮本 良磨 Yoshimaro Miyamoto

日本アイ・ビー・エム(株)  
システム製品事業テクニカル・セールス  
エグゼクティブITスペシャリスト



山本 隆之 Takayuki Yamamoto

日本アイ・ビー・エム(株)  
GTS事業スマーター・クラウド事業部  
コンサルティングITスペシャリスト



岩村 郁雄 Ikuo Iwamura

日本アイ・ビー・エム・システムズ  
エンジニアリング(株)  
ビジネス・プラットフォーム推進  
サービスマネジメント・エバンジェリスト



増田 みさお Misao Masuda

日本アイ・ビー・エム(株)  
GTS事業ITSデリバリー  
ITアーキテクト



中井 秀有 Hidekuni Nakai

日本アイ・ビー・エム(株)  
テクニカル・リーダーシップ  
プロフェッションズ開発

※本コラムは、IMO(ITSM Management Office)のメンバーにより執筆されています。IMOは、ITサービスマネジメントおよびITILの活用促進と日本での展開への貢献を目的として、日本IBM社内に2004年に設立されたバーチャル組織です。

### 【参考文献】

- [1] 一般社団法人 日本情報システム・ユーザー協会:第17回企業IT動向調査2011(10年度調査)~ユーザー企業の現状と課題~. [http://www.juas.or.jp/servey/it11/it11\\_summary.pdf/](http://www.juas.or.jp/servey/it11/it11_summary.pdf/)
- [2] サービストランジション, TSO(2008).
- [3] IBM: Tivoli Unified Process. <http://www-06.ibm.com/software/jp/tivoli/products/unified-process/>