

システム運用「人としくみ」

これまでのシステム運用

コンピュータを利用し始めた当時は開発とか運用という言葉はありませんでした。しかし、業務量やコンピュータで処理すべき業務種類が増えていくにしたがって、開発や運用という仕事の役割を分担するようになったのです。そして、なぜかシステム運用には「開発からの引継ぎ」という言葉が生まれてしまいました。

業務システムの開発と運用

もともと業務システムの開発とシステム運用では、それぞれ求めるものが異なります。生産性、効率性、コスト、品質など、言葉は同じでもそれぞれの中身には大きな違いがあります。たとえば効率性を例にあげてみます。

システム開発の効率に対する考え方は、いかに少ない工数で、しかも、いかに短期間で開発を終えるか、ということが最優先になります。したがって、開発するプログラムやステップ数など、開発ボリュームをできるだけ少なくするようにします。このような前提に立ちますから、たとえば、何らかの判断を必要とする要求仕様に対しては、プログラムにロジックとしてもたせず、人手などの安易な手段を採用するケースもでてくるわけです。

こうしたことに対して、システム運用の場合はできるだけ人手の介入を少なくしたいと考えます。そのためには、同じような処理プログラムがあっても運用上の判断が少ない方がよいと考えます。人手が入るということは、属人性が増す可能性が高く、人為的ミス誘発にもつながります。システム運用は安定性・安全性をもっとも重視しますから、これを阻害する要因はできるだけ避けなければならないのです。

コンピュータが企業の業務システムに利用されはじめた頃、そして、それからの長い期間は業務システムを開発した技術者たちが、当然のようにシステム運用も行っていました。開発したものを自分が運用する。これは当時のコンピュータ技術者にとって当たり前のことだったのです。したがって、頭で考えたことを実際に運用してみると、頭だけでは理にかなわないということをよく理解できたものです。

当時のシステム開発技術者であっても、基本的には開発業務に専念したいと考えてい

ました。しかし、自分たちが開発したものをしっかりと運用しなければならない立場にもありました。開発業務に多くの時間を割くためには、いかに運用を楽にするか、つまり、いかに手のかからないものにするか、いかにミスを犯さないようにするかを徹底して考え、こうしたことをルール化して、これに反する事項はすぐにもでも改善しなければなりません。これが「運用改善」のはじまりだったのです。

たとえば、月初に比べ月末になると処理時間が数倍にも達してしまうという問題をどう解決するか、ということも運用改善の一つでした。これは、処理データ量に比例して処理時間が延びるというシステム(プログラム)設計上の問題でもあります。このようなシステム改善も運用改善として行っていました。そして、こうした一つひとつの改善を、開発に、運用にと、それぞれのガイドラインやルールを定め、フィードバックしていったのです。

しかし、業務システムの開発が急増し始めたころから、だんだんと開発・運用の役割分担を明確にせざるを得なくなりました。業務システムの開発規模が大型になってきたことも、そうしたことを加速させました。そして、1985年ごろにはそれがより明確になってきました。このキッカケは、EUC(End User Computing)が始まったことと、業務の処理量(業務システムおよびデータ)が急増し始めたこと、新技術が競うよう出現しはじめたことなどでした。この1985年というのは、企業ネットワークの整備がほぼ終わり、ネットワークコンピューティングが本格的に動き始めた一つの区切りの時代であったように思います。

EUCは、業務部署が自部門主導で業務システムの開発を行い、自部門に中小型コンピュータを設置して業務処理を行なうというものです。コンピュータが小型化になり、特別な空調設備などを必要としなくなったことが、このEUCの普及を早めたと考えられます。また、費用対効果をより明確にするためにも、情報システム部門が全社予算を確保しそれぞれの業務部署に割り振るよりも、自部門の責任で予算を確保して業務システムの開発・運用を行なうべきとの考えも当時は有力でした。

さらに当時は、システム運用のさまざまな制限・制約に対する業務部署の反発もありました。システム運用部門は、自部門の責任範囲にあるすべての業務システムを対象に、もっとも効率的な運用を目指します。そのため、時には、さまざまな制限・制約を業務部署に強いることとなります。たとえば、オンライン・サービスの時間帯を9:00~18:00までという制約などです。この時代の背景からしても、企業競争が激化する中で、このような制限・制約は業務部署にとっては我慢のならないものとなっていきました。“この部門のこの業務はこうしてほしい”との要求が相次ぐようになり、それを認

めないシステム運用部門と要求元である業務部署との対立が続くようになりました。こうした時期に、企業内ネットワークというインフラ整備がほぼ終わり、簡易設備で設置できる中小型コンピュータが出現したのです。

さらに、この時代、企業では、業務のシステム化に拍車がかかり、業務システムは“産めや増やせや”の時代でした。一種のシステム開発ブームでした。そこで、情報システム部門が管理するボリューム(業務システム・データの量)が急増しはじめ、設備増強やパフォーマンスアップのためのチューニング作業などに多くの工数を割かざるを得なくなりました。

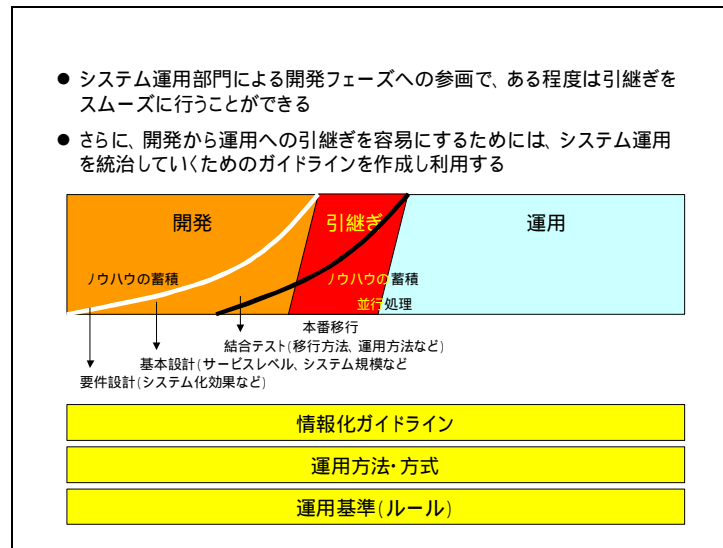
いままでとは異なる新しい技術も出はじめました。システム開発は、その時の最新技術を用いることが多いものです。新しい技術の採用は、業務システムを利用する人たちや開発技術者をより満足させることができます。使い勝手がよく、生産性が高くなり、作業効率を向上させるからです。また、いちど開発した業務システムは、5年、10年、15年と使い続けられるわけですから、できるだけ新しい技術を採用しておく必要もありました。しかし、新しい技術を採用した業務システムの開発には、新技術の習得や開発ルールの改定など、それ相応の負担が必要になりました。

このような背景から、システム開発、システム運用、それぞれに新しい取り組みが必要となり、全体効率を考えると、開発と運用という2つの役割分担をせざるを得なくなったのです。開発と運用の役割分担が行なわれるようになってから、システム開発部門は、より開発の効率性を追及するようになり、システム運用部門は、運用の効率性やサービスの追及を強化しはじめたと考えられます。

また、システム開発の外部への委託やSIベンダーへの丸投げなども多くなり、業務システム開発プロジェクトへのシステム運用部門の人たちが参画することも少なくなってきました。このように、だんだんと、システム開発とシステム運用の2つの部門には断層ができるようになって、開発は開発、運用は運用、というような意識が強くなり、開発が終わったら運用に引き渡すという構図ができ上がってしまったのです。

業務システムを運用するには、別紙(表-2)にあるような多くの運用要件を整備しなければなりません。開発が長期間かけて開発した多くのものを、短期間で運用に引き継ぐことは不可能なことです。開発時には、運用に関する多くのノウハウも蓄積されていますが、これらがすべて書類に記述されているとは限りません。しかも、膨大な書類だけで引き継ぐこともまた現実的ではありません。そこで、図-9に示すような、システム運用部門の開発フェーズへの参画と運用ガイドラインの制定が重要にな

ります。



(図 - 9) 開発から運用への引継ぎ

この図 - 9 は、別紙 (表 - 2) にあるようなシステム運用要件を、できるだけ早い段階に入手するとともに、業務システムの開発にこの運用要件を組み入れるべきであることを説明しています。また、すべての業務システムに該当する運用要件は、情報化ガイドライン、運用方法、運用基準などで標準化し、できるだけ統一されたシステム運用が行なえるようにすべきことを表しています。

最近では、各企業で、情報システムに関する日本版 SOX 法への対応を進めていますが、この内部統制(IT 統制)の面からも、情報システムの開発と運用の役割を明確に分割することが求められています。このようなことから、情報システムのスムーズな運営を行なうためのしくみの重要性がさらに増してきています。

(別紙 表 - 2)

運用業務	作業項目	代表的な作業内容
運用設計	キャパシティ調査・調整	現状リソースの実績把握基準 業務システム導入時のキャパシティ調査基準 課金・配賦基準 キャパシティプランニング基準 など
	ジョブ構成基準の作成	バックアップ/リカバリー基準 単位あたりのジョブ実行時間 ジョブ連携基準 ファイル連携基準 帳票出力基準 など
	サービスレベルの保証方法	S L Aの内容 個別実績の集約基準 情報公開基準 S L A評価基準 S L A評価会議の開催基準 など
	業務システムの移行方法	新旧並行処理の要・不要 ファイルやDBの移行方法 バックアップ計画 バックアップ資材(媒体)購入計画 移行計画 など
	要員計画	業務システム導入後の運用要員体制 業務分担の変更 連絡ルート体制の変更 各種マニュアルの整備 要員教育 など
運用開発	バックアップ・ポイントの設定	再処理可能なバックアップのとり方 単位あたりのジョブ実行時間に合わせたバックアップ・ポイントのとり方 など
	ジョブ、ジョブネットワークの構成設計	処理タイミングに合わせたジョブとネットワーク構成 中間ファイルや連携ファイルの出力時期 など
	帳票出力などに関する情報集約	帳票名 標準在庫量 発注リードタイム 出力先条件情報 など
	各種運用情報のマスター登録	J C Lやシエルの構成情報 など
	ファイル、DB、プログラムの移行作業など	それぞれの一覧表 保管場所 移行時期 など
緊急時対応	業務システムの品質保証と障害時の緊急対応	障害種類の明確化 不正データの発見方法、影響への対応 障害の早期発見のための設備やしくみ など
	障害発生時の緊急対応ルール	連絡ルート、体制 標準的な再処理時間の取り決め など

(表 - 2) 運用要件の一例