

3.27 基幹系システムにパッケージソフトを適用する際の教訓(その1) (T27)

教訓
T27

パッケージはサポートを買え

問題

A社で保有する社内特定業務向け基幹業務システムは、現場部門に対して24時間のオンライン業務サポートを実施している。業務は日中だけでなく夜間も継続しており、かつシステム停止による現場業務への影響が大きいことから、運用サポート部門も24時間体制でシステムの運用をサポートしている。業務システムは当該用途向けの海外製パッケージを購入して構築されており、アプリケーションサーバ、DBサーバ、他システムとの連携サーバなど各層のサーバは冗長構成になっている。構築したシステムは本稼働後、2カ月目を迎えようとしていた。

ある日の午前、アプリケーションサーバが突然システムダウンした。そこから、システム回復までの経緯はおおよそ以下のとおりである(カッコ内の時刻はパッケージ開発元の現地時間を表す)。

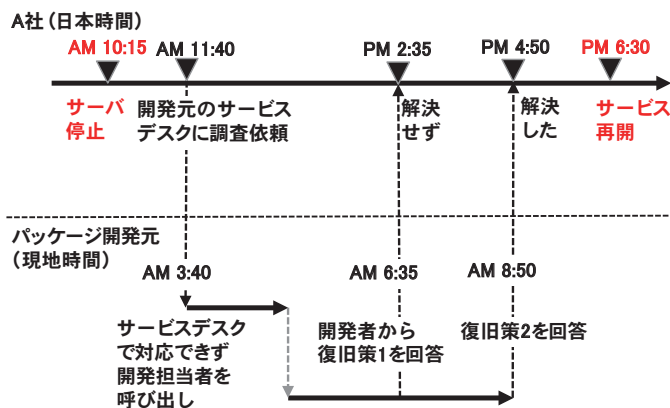


図 3.27-1 障害発生後の対応状況

- 10:15 (2:15) 運用センターでアプリケーションサーバのシステムダウンを検知
すぐにサーバ再起動、スタンバイへの切替えを行ったが状況は変わらず
- 11:40 (3:40) 導入したパッケージの開発元であるB社のサービスデスクに調査を依頼
その際にサーバのシステムリソースの枯渇状況も連絡
サービスデスクでは問題を解決できず、パッケージの開発チームを深夜に呼び出す
- 14:35 (6:35) B社から復旧策1を受領
指示された作業を実施したがサーバダウンは解消せず

- 16:50 (8:50) B社の継続調査により、原因と思われるデータの削除を提案される(復旧策2)
提案の作業を実施した結果、サーバダウンは解消した
- 17:30 (9:30) 他システムとの連携を再開し当該システムへのデータ入力を再開
- 18:30 (10:30) すべてのデータ入力が完了し正常稼働を確認した上で、サービスを再開

問題発生後、サービス再開まで8時間を要した結果、その間、当該特定業務の業務運用は手作業に頼らざるを得なくなり、当該業務を通してエンドユーザーに提供しているサービスにも大きな影響が出た結果、その復旧を含めてA社のビジネスに大きな損害をもたらした。

原因

サーバがダウンした直接の原因は、導入したパッケージの潜在的な不具合によるサーバのシステムリソースの枯渇(アプリケーションが使用する内部テーブル上に古い情報が滞留したこと)であり、A社で初めて発生した事象であった。このパッケージはA社以前に複数社に導入された実績のあるものであるが、今回のトラブルが先行する他の導入先で発生しなかった理由は、先行する他の導入先では取り扱うデータ量がA社ほど大量ではなく、ディスク、メモリ等のシステムリソースを枯渇するまで使い切ることがなかったためであった。A社がこのパッケージで取り扱うデータ量が他の先行する導入先に比べて圧倒的に多かったことが、この障害を初めて顕在化させた。

また、このトラブルの発生後、復旧までに長時間を要した理由は、A社からB社に調査を依頼する先がB社のサービスデスクであり、開発チームがすぐに調査を開始することになっていなかったため、高度の技術、製品に対する豊富な知識を必要とする調査であることの判断、深夜に開発チームを呼び出して調査を開始するまでの初動に時間を空費したことがあげられる。

対策

上記の原因を特定したA社では、直接の原因であるパッケージの不具合の修正だけでなく、今後、同様にトラブルが発生した場合に、対応時間を短縮するために、以下の対応をB社と協議の上で実施した。

【対策1】B社開発チームによる24時間サポート体制の構築

このシステムで提供する業務は障害発生時には回復最優先であることから、パッケージ開発者自身による調査を迅速に開始することが最も効果がある施策であると判断したA社は、障害発生時のB社連絡先をこれまでのサポートデスクからB社開発チームのメンバへの直接連絡に変更し、日本国内製の他の重要パッケージのサポートと同様に24時間いつでも直接開発担当者によるサポートを受けられるようB社に対応強化を依頼した。その上で、B社の開発チームのメンバの氏名を公開してもらい、当番制で直接連絡できるようにした。

【対策2】 トラブル対応のサービスレベルを明確化

トラブル発生時のファーストコールから回答までの目標時間を設定し、この時間内に正しい回答が得られない場合にはペナルティを課すことを SLA (Service Level Agreement) に定め、サポート契約を締結し直した。

これを今回のトラブルに当てはめると、下記のようになる。

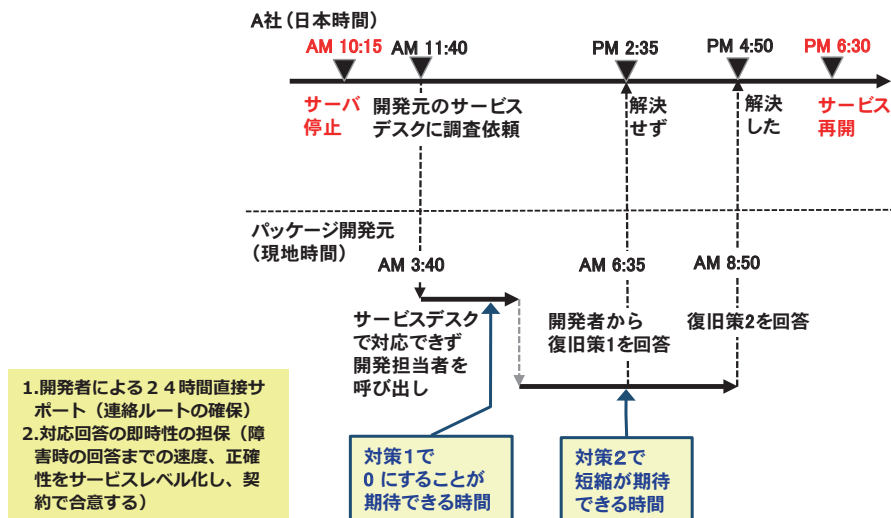


図 3.27-2 施策と期待効果

効果

上記の施策により、この事例では以下の効果を得た。

(1) トラブル発生時の回復までの時間短縮

この施策で期待する効果そのものであり、A社ではその後発生した別内容のトラブル対応において大幅な時間短縮(再開まで1時間)を達成し、施策に効果があることを立証した。

(2) SLA 締結による調査精度向上

SLA 締結はB社内での緊張感の持続をもたらし、結果として後続の障害における調査、対応方法提案の精度を向上させることができた。その効果は上記(1)の時間短縮に表れている。

教訓

この事例からは、一時的なシステムの停止が業務に大きな影響をもたらすシステムにパッケージを組み込む際の以下の注意点が教訓として活用できると考えられる。

(1) パッケージ提供元の開発チームによる直接の協力体制の構築

業務時間中にシステムが停止すると業務に多大な影響を与えるようなシステム(ITサービス)にパッケージを組み込む場合には、そのパッケージが運用中にトラブルを発生させた場合に即時対応でき

る体制を開発元の協力により構築することが有効である。自社開発したシステムにおいて、開発に関わったチームを維持メンバとして保持することに相当する行為として、パッケージの内容を最も知悉した開発元の要員に素早く調査を委ねることが最速のトラブル解決方法になる。

(2) トラブル対応の目標設定による協力と緊張の維持

トラブルに即時対応する体制を構築しても、ゴールを定めないと、調査の開始時間が早まるだけで、調査時間の短縮には直接は結びつかない。

そのため、今回の事例のように、トラブル対応の委託先とは以下のような事項について目標を設定してSLAとして文書化し、契約することが有効に作用する。

(SLAの項目の候補)

- トラブル発生との連絡が開発元の技術者に受け付けられるまでの時間
- 調査・回答の提案までの時間
- 回答の正確性(回答によって事象が解決した率)
- パッケージ障害発生率

これらの事項を決定して契約することにより、開発元にも緊張感が生じ、製品やサポートの品質に対する意識の向上も期待できる。

(3) 重要なシステムリソースの枯渇監視を組み込む

今回の事例でもサーバリソースの枯渇監視を実施しており、本件の原因になったリソースの枯渇が報告されていたが、枯渇の原因まではわからず、解決には開発元の調査を待つしかなかった。だからといって、リソースの枯渇監視には意味がないというわけではなく、リソースの枯渇状況は開発元に情報として伝えられ、調査の取り掛かりに有力な情報になった。直ちにトラブル解消に結びつかなくても、予兆の察知や調査情報の提示に有効に働くことが期待できることから、重要なサーバのシステムリソースの枯渇監視も実施を検討すべき施策であると考えられる。

昨今、システム構築においてパッケージ製品を組み込むことが多くなってきている。パッケージを適用することは、システム構築のスピードを向上させることに多大なメリットをもたらすが、同時に内容がブラックボックス化して導入した会社だけではトラブルを解決できないというデメリットをもたらす。システム構築、特に高度の安定稼働を必要とする業務システムの構築にパッケージを適用する際には、パッケージ本体の製品品質だけでなく、何かあったときに必要なサポート体制を高い品質で提供してくれる開発元であるかどうか、パッケージ選択の重要な要素になる。この教訓では、パッケージ選択の際には、適用するシステムの特性に応じた必要十分なサポート体制を供給してくれる提供元であるかどうかを確認することが重要であること、国内はもちろんであるが海外製パッケージを導入する際には、現地時間の休日・夜間においても上記のサポート体制を維持できるかどうか、導入の重要な決め手になることを発信したい。

教訓タイトルは、上記を強調することを狙いとして「パッケージはサポートを買え」とした。