

銀行基幹系システムのKLI策定と実践

溝田 聡 茂木 大地

Defining and Using KLIs for Mission Critical Banking Systems

Satoshi Mizota and Daichi Mogi

企業に導入された IT 資産の中には使用開始後のビジネス環境の変化に対応するため、当初の資産に継続的な保守や拡張を実施し、長期にわたって利用されているものがある。一般的にレガシーと呼ばれるこれらのシステムをブラックボックス化させずに、ビジネス価値を生み続ける資産として維持することは企業にとって重要である。金融機関のお客様を担当している IBM の IT アーキテクトによるタスクチームは、銀行勘定系オンラインとして 20 年以上使用されている Data Systems Environment (DSE) バンキングシステムとその環境に対して、レガシー化度の評価項目として KLI (Key Legacy Indicator) を定義し、それに基づいた評価作業を実施した。当論文ではその経験による KLI 策定方法、それを使った評価の実践方法、その有効性について論じる。

Some corporate systems are repeatedly maintained and expanded and kept running over a long period. These systems are generally called legacy systems. It is extremely important to prevent legacy systems from becoming black boxes and to ensure that they continue to produce business value. An IBM IT architect group working with financial customers has defined KLI (Key Legacy Indicators) and evaluated the DSE (Data Systems Environment) banking systems that have been used as the account processing systems in Japanese banks for over two decades. This paper describes the methodology of defining and monitoring KLIs and describes the overall effectiveness of KLIs based on our practical use of them.

Key Words & Phrases : KLI, キー・レガシー・インジケーター, 銀行, 勘定系, DSE バンキング

KLI, key legacy indicator, banking, mission critical system, DSE banking architecture

1. はじめに

IT 業界ではレガシーの言葉をさまざまな意味で用いている。重要で長期に使用されているコアなソフトウェア資産であるとの意味であったり、長期に改善が放置されブラック・ボックス化したのが完全に捨てることのできない古いシステムであったり、その意味の差は大きい。本論文でのレガシー化は前者ではなく後者の意味である。

Working Effectively with Legacy Code [1] においてレガシーコードとは、単にメインフレームのアプリケーションのことではなく、変更することが困難なコードであると定義している。プログラムソースやデータ項目は、更新・追加を続けることによって重複部分や複雑性が増し、保守の容易性が開発当初と比べて低下してしまう。そこで SOA 化による既存システム資産の再利用性や相互接続性の向上が可能となってきている中、レガシー・システムを段階的に改良していくアプローチが見直されている。

IBM の 2010 Global Technology Outlook (GTO)^{*1} の中では“Future of Legacy”に着目し、KLI (Key Legacy Indicator : キー・レガシー指標) とレガシー・コントロール・ループを提唱している。KLI は、企業の IT システム群の中でレガシーと位置付けられる部分に対して、ヘルス・チェックを行うために使われる指標である。また、レガシー・コントロール・ループは、KLI を使ったヘルス・チェックをする「評価・識別ステップ (Identify & Qualify)」、識別されたレガシー・システムの負の部分改善する「改善ステップ (Improve)」、それを適切に運用する「運用ステップ (Operate)」の大きく 3 つのステップで構成されている。これらのステップを継続して繰り返すことで、レガシー・システムを最大限に有効活用しようとするものである。しかし、残念ながら具体的な指標と方法論は示されていない。

一方、社内の金融機関担当 IT アーキテクトにより構成されるタスクチームは、Information Management System

提出日:2010年9月6日 再提出日:2011年3月11日

*1 IBM の基礎研究部門が中心になって 1982 年以降毎年まとめている技術動向情報

(IMSTM)^{*2} や IMS/ESA[®] オンライン適用業務開発／運用支援プログラム (SAIL/ESA)^{*3} を中心とした銀行基幹系システムのアーキテクチャーである Data Systems Environment (DSE) バンキング [2] を今後 10 年以上継続使用していくために必要な施策について検討を行ってきた。その中で日本の銀行基幹系システム向けに KLI を定義し、レガシー・システムと位置付けられる DSE バンキング・システムのヘルス・チェックを行うことにより、継続運用に必要な改善策の優先順位をお客様に提示するきっかけにしたいと考えた。

本論文は、タスクチームが行った KLI 定義の方法と実際に KLI を使用した評価作業を実践した結果得られた知見について解説する。

2. 銀行基幹系システム向けのKLI策定の目的

KLI を用いたヘルス・チェックの用途はお客様のシステムの環境や状況により臨機応変に利用できるものである。ここでは銀行基幹系システムに対する活用方法として筆者らが狙った 3 つの例を挙げる。

2.1 レガシー化度合いの自己評価

1990年代に第3次オンライン・システムとして続々とサービスインした銀行勘定系システムは、初期開発以降もさまざまな新商品・新サービスに対応するために多くの開発を積み重ねてきた。それに伴い既存コードに対するロジックの追加や、既存プログラムのコピーによる新規プログラムの作成が繰り返されている。リーマンの法則 [3] の中で述べられているように、システムは特別なことをしないかぎりその複雑度は増加し続けるため、現在のプログラムコードは一般的にサービスイン当初と比較して保守性が損なわれている。また、開発に携わった担当チームの特性、開発時期、プロジェクト期間、投下コストなどの諸条件により、アプリケーションやデータベースのレガシー化のレベルも勘定系システム内でさまざまである。

KLI の評価を勘定系内のサブシステム単位（流動性預金・定期性預金・融資・為替・対外系など）で実施することで、どのサブシステムにおいてレガシー化がより進んでいるかを明確にすることができる。レガシー化が進むと開発生産性やプログラム品質に対してマイナス面の影響を与える可能性が高まる。勘定系システムを一体で考えるのではなく、サブシステムやモジュール群の中で劣化した部分を識別し、優先的に改善プロセス（レガシー・コントロール・ループ）を実施していくことで IT 投資効率を高めることができる。

2.2 自行システムのウイーク・ポイントの明確化

レガシー化が進むことで、アプリケーション開発やシステム保守に課題が生じるのは勘定系システムだけではない。営業店端末やコールセンターなどのフロント系システム、データ・ウェアハウスや CRM などの情報系システムも銀行にとっての基幹システムであり、レガシー化によるさまざまな非効率化を防止する必要がある。オープン基盤上でオブジェクト指向型言語を用いて開発したシステムであっても、サービスイン後のアプリケーションの改修が続くことにより、プログラム・コードの乱れやデザイン・ドキュメントとの不整合などの課題が徐々に増えていくのはホストシステムと同様である。

KLI を用いて銀行で保有しているシステム群を横断的に評価することにより、レガシー化対策が必要なシステムを識別することができる。またそれぞれのシステムにおける品質と生産性の相関やレガシー化への対応レベルを比較することにより、一般的に運用コストが高いと見られている勘定系システムの投資効率を正しく評価することができる。

2.3 進ちよく効果の評価、目標設定

どのようなシステムでもサービスインした瞬間からレガシー化の道を歩み始める。従ってシステムに対する効率改善は継続性が重要である。まずシステムやサブシステムの中でレガシー化が進んでいる部分を探す「評価・識別ステップ」から始める。KLI はこのステップで使われるべき評価指標である。

評価作業の結果に基づいてさまざまな改善を行うのが次の「改善ステップ」である。レガシー化したアプリケーションや基盤に対して、冗長性を排除したり、開発プロセスを改善したりすることで、レガシー・システムを長期に渡って使い続けることが可能になる。そして次の「運用ステップ」では人的資源やハードウェアを維持しながら、ビジネスに必要な不可欠なレガシー・システムを運用・管理していく。

この 3 つのステップを繰り返しであるレガシー・コントロール・ループの中で「識別・評価のステップ」を定期的に行い、KLI 指標値の変化をチェックすることで、改善施策の効果を把握し、新たなレガシー化発生の兆候を監視することができる。また、KLI 指標そのものを来期の改善計画の業務数値目標として設定することも可能である。

このように KLI の用途は多様であり、それぞれの目的

※2 ホスト DB/DC 向けミドルウェア

※3 金融機関向けオンライン・アプリケーション制御プログラム

や対象システムの環境に応じて評価項目をカスタマイズすることが可能である。重要なことは一部の事象や思い込みによる決め付けではなく、システムのレガシー化を相対的かつ客観的に判断するための評価軸を設け、効果的な改善策につなげていくことである。

3. インデックス項目の定義

KLIを実践的に使うためには、まずヘルス・チェックの目的を十分に想定した指標項目の定義が重要である。銀行基幹系システムのKLIは、DSEバンキング・アーキテクチャーやオンライン・アプリケーション・プログラムの構造に対して評価を行い、このアーキテクチャーを使い続けるために必要な改善分野をお客様自身が見つけて納得するためのものである。

評価は、システムを開発・管理・運用されている当事者に依頼するため、評価項目は分かりやすさを重視する。また評価を短時間(30分程度)で実施可能とするために、全体の項目数は30から40程度が望ましいと判断した。

3.1 大項目の選定

KLIではQuality Inspection (QI)^{*4}手法による品質評価の実施原則と同様に、全体からのトップダウンアプローチにより全体像を把握しやすくする。評価者が実際に実務でかかわりを持ち、かつ問題意識を持っている分野に絞り込み、大項目とする。この大項目ごとの平均評価値でシステムのもつレガシー化度合いを相対的に把握できるようにする。

銀行基幹系のレガシー化の評価軸として次の6項目を選定した。業種、対象システム、実施目的によって選定される項目は違ってくる場合がある。

- (1) アプリケーションに関するレガシー
- (2) データに関するレガシー
- (3) 他システム連携に関するレガシー
- (4) IT基盤に関するレガシー
- (5) 開発／テストに関するレガシー
- (6) 人材育成に関するレガシー

この6項目を選定したのは、われわれの議論から項目を厳選するなら、Model, View, Controller (MVC) 視点により、データ、外部インターフェース、プログラムのレガシー化度に着目し、さらにそれを支える基本要素として基盤、開発手法、人材とすべきと判断したことによる。また、この6項目はお客様からもよく言及される項目であった。

このほかに評価軸となる大項目の候補として「ビジネス・プロセス」、「システム運用」などが挙げられたが、銀行基幹系システムのレガシー化度の評価を目的とした場合、6項目に代表されるコア部分と比べて二次的な項目であると判断し項目から除外した。カスタマイズするときも評価者の情報量やスキルにより判断が難しい分野やヘルス・チェックの目的と直接関係しない項目などは思い切ってKLI対象から外すべきと考えている。

3.2 中項目へのブレイクダウン

大項目として選定した分野に対して、中項目の評価軸にブレイクダウンしていく。数値で示しやすいものを選択し、

表 1. KLI 中項目

1. アプリケーションに関する KLI	
1-1	コンポーネント化 (サブプログラム化, マクロ化など) による再利用性
1-2	重複コード, デッドコードの度合い
1-3	構造化プログラミング
1-4	コーディングの標準化
1-5	ドキュメントの整備
1-6	生産性
1-7	品質
2. データに関する KLI	
2-1	データ・モデルの維持
2-2	データ定義
2-3	重複データの度合い
2-4	物理DBの維持・保守
2-5	変更対応の容易性 (拡張性)
2-6	データの正確性
3. 他システム連携に関する KLI	
3-1	チャンネル・インターフェースの柔軟性
3-2	チャンネルとアプリケーションの非依存性
3-3	システム間連携基盤 (ハブ, ESB など) の柔軟性
3-4	チャンネル/端末の情報管理・運用管理
4. IT基盤に関する KLI	
4-1	ハードウェア・サポート
4-2	ソフトウェア・サポート
4-3	ベンダーによる高信頼性システム技術支援
4-4	研修, 情報提供
4-5	運用管理の共通化, 統合化
5. 開発・テストに関する KLI	
5-1	ツールによる開発
5-2	テストの自動化
5-3	トレーサビリティ
5-4	ライブラリー管理
6. 人材育成に関する KLI	
6-1	IT部門要員の充足
6-2	人材のローテーション
6-3	スキル
6-4	キャリア形成計画
6-5	システム・デザイン・コンセプトの継承

*4 プログラム品質検証の手法

総数が多くなり過ぎないように大項目あたり4項目から7項目を基準とする(表1)。

3.3 評価インジケータ

KLI 中項目に対して1~5の5段階で評価を依頼する。本来、評価値は客観的なデータから単純判断ができるものが望ましい。しかし、数値データを持ち合わせていない場合に評価者が調査することにかかる負担を考慮し、優れている/平均的だ/劣っているなど、主観的な判断ができるようにした。ただし、あまりにも評価にばらつきが発生しないように、各中項目のインジケータ値に対して目安の説明文を加えた。

説明文は表2のようにインジケータ値1, 3, 5の3段階に対して行う。3を一般的なシステムであれば満たしているレベルとし、著しく優れている場合は5、レガシーの兆候があり、大幅な改善を要するものを1とした。実際に評価を行うと大半の評価が3から4の間になり、一部がそれ以上ないしは以下になることが望ましい。レガシーシステムの評価作業自体に否定的な印象があるため、平均評価値としては中央値の3よりやや高めに出るように設定し、前向きな評価作業ができるように配慮するためである。

表2. KLI 評価インジケータの例

	KLI 中項目	評価インジケータ
1-1	コンポーネント化(サブプログラム化, マクロ化など)による再利用性	5: コンポーネント化(サブプログラム化, マクロ化など)に対する標準化が徹底されており, 再利用性が非常に高い 3: コンポーネント化に対する基準が設定されており, 共通利用されている 1: 再利用できるコンポーネント(サブプログラム, マクロなど)が非常に少ない

3.4 参照メトリクス

中項目で挙げたものには、さまざまなメトリクスにより数値化が可能なのが存在する。お客様がインジケータに対して客観的な測定値を望まれた場合や、重点課題分野に対する詳細分析が必要な場合に備えて測定メトリクスの候補をあらかじめ列挙しておくとい。

金融基幹系システムに対するKLIでは、アプリケーションや開発・テストの大項目エリアに対して、参照メトリクスとしてIEEE Standard Dictionary of Measures to Produce Reliable Software [4]に記載のメトリクスを対応づけた(表3)。

3.5 レーダー・チャート

図1に大項目単位で評価値の平均値を集計し、レーダー・チャートとして表す。対象のシステムに対して複数

表3. KLIと参照メトリクスの対応例

KLI 中項目	参照メトリクス
1-1	15. アーキテクチャーのグラフィック複雑度 16. 循環的複雑度
1-2	
1-3	16. 循環的複雑度 25. フローの複雑度
1-4	
1-5	9. ヒューマン・エラー分布図 11. 欠陥検出時間 19. 詳細設計の構造 32. ソフトウェア文書とソースコード
1-6	
1-7	1. 障害密度 4. 障害総日数 6. 欠陥指数

の評価者毎のレーダー・チャートを作成する。一般的には同じお客様のシステムに対してレーダー・チャートは類似した形を示すと考えられる。

KLIはそもそも客観的な測定値を用いたものではないが、レーダー・チャートの形状を比較することで他行や他サブシステムとのベンチマークも可能となる。

銀行勘定系システムに関するキー・レガシー・インデックス

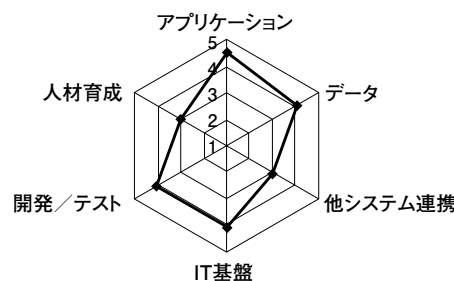


図1. KLI レーダー・チャート

4. KLIによる評価の実践

KLIによる評価にあたっての考慮点、有用性を確認するために、実際に7地方銀行の11名のお客様および5名の弊社担当者が評価実施した。このフィールド・テスト時の結果と評価者からのコメントも踏まえ知見を導出した。

4.1 評価者の選定

KLI 評価者の選定にあたっては、基幹系システム全体の総合的な評価を実施できる方、組織としての方向性・方針を検討できる方、問題認識を持っている方などにご評価を依頼することが望ましい。評価者の選定にお

いては、評価結果活用の観点から、次の3点の評価者の立場を考慮することが必要である。

(1) システム部長、副部長など部門管理者による評価実施

部門管理者の方であれば、組織的に継続的なPDCAサイクルを回せるという観点から、組織全体としての強み、弱みの認識とともに、セルフアセスメント可能なKLIを改善指標としてご利用いただくことが可能となる。

(2) チーム・リーダーによる評価実施

チーム間での評価内容を共有できるという観点から、複数チームのリーダーの方で実施し、各チームでの強み、弱みをボトムアップ的にシステム全体、組織全体としての評価につなげていくことが可能となる。

(3) チーム・メンバーでの評価実施

チーム内での問題認識を共有できるという観点から、チーム・リーダーとチーム・メンバーで実施し、具体的な評価項目における改善策を検討することが可能となる。

4.2 評価の依頼と実施に関する考慮点

評価依頼の際には、この評価をどのような目的で実施しようとしているのかを明確にすることが必要である。弊社が考える目的については、銀行基幹系システム・アーキテクチャー白書 2010 [5]にも記載があるので、参考の一助とされたいが、お客様における実施の目的を、一度再定義されることをお勧めする。それは、これから組織的な改善プロジェクトをスタートさせる判断とするというような大きな目的でなくとも、現状の評価を目的とすることもよい。実際の評価実施については、目的などを実施要綱にまとめ、アンケートとして実施することが効果的である。内容的に理解が難しい項目はないため、一度、お客様自信で評価項目をご確認、仮評価いただく。その後、お客様にご判断に困られたインジケータの意味を補足ガイドし、コメントなどをヒアリングしながら評価の確定を実施する。これによりお客様から評価項目の納得感を得た上で、評価を固めていただくことができ、かつ、お客様の考え、考慮点を具体的に確認することができる。フィールド・テスト結果からは、次のことを考慮点として確認した。

(1) 事前説明時の考慮点

- ・KLIの評価インジケータは、あくまでもKLI中項目を分かりやすくするための補足説明であるので、各評価者に具体的にイメージする点があれば、それをインジケータとして利用することも可能であること。
- ・1, 3, 5と代表的な評点を示しているが、2, 4や3と4の間で3.5といった評点も可能であること。
- ・評価できない分野、項目は、一時的には評価しないことも可能であること。

(2) 事後ヒアリング時の考慮点

- ・評価者がイメージされたインジケータの内容と評価理由を具体的に確認すること。
- ・未評価項目においては、KLI中項目での代替インジケータ候補のガイドにより再評価いただくことも試みる。

4.3 評価結果の考察

フィールド・テスト結果では、同じ銀行の基幹系システムを複数の担当者で評価した際に、評価者が思い浮かべる対象が異なることや、評価基準も評価者の主観的判断となり、評価が異なる場合があるが、大項目レベルでは特徴を捉えることが可能であることがわかった。以下、評価結果から得られた特徴的なレーダー・チャートを例示して考察を加える。

(1) KLI評価インジケータの有効性

図2は、A銀行での部門管理者2名の方による評価結果レーダー・チャートである。2名が協議して評価した事実もなく、同レベル管理層であればおおむね評価が一致した。評価者が少ないため、これだけで断定することはできないが、KLIの評価項目の評価インジケータの内容の有効性を示している。

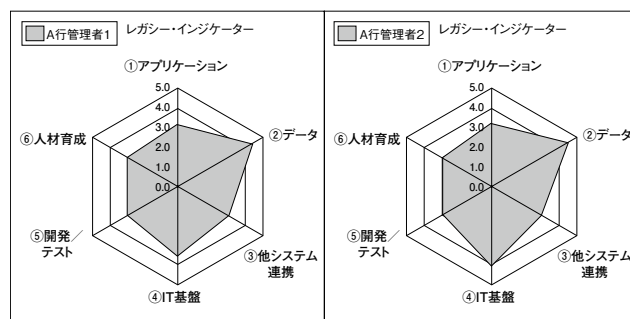


図2. KLI インジケータ有効性検証

(2) KLIによる相対評価

図3は、B, C, D, E銀行の評価結果レーダー・チャートである。

これら、4行とA銀行の評価結果とも合わせ考察する。A銀行とB銀行は、おおむね相似形となっているが、各銀行とも、基幹系「データ」の維持、管理については、相対的に高い評価をしている。また、IT基盤やアプリケーションにおいても、高い評価となっている。これは、第3次オンライン以降から今まで、長期間にわたり基幹系を維持、拡張、保守してきた実績からの評価であると考察される。これと比較して、「人材育成」や「他システム連携」については、相対的に低い評価となっている。一部

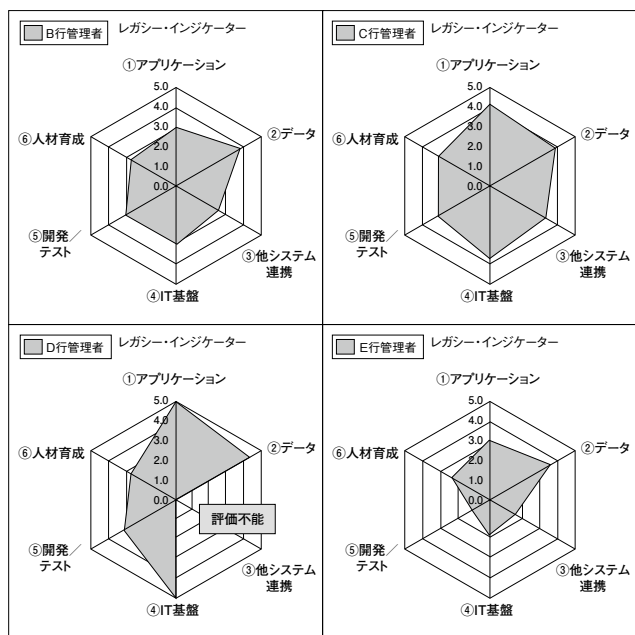


図 3. KLI インジケータ－相対評価

インジケータ－に記載された評価レベル 5 記述の実現性の困難さもあるが、地方銀行の抱える問題、課題の存在が可視化された。人材育成面では、スキル継承の問題、ローテーション、キャリア形成の問題、課題があり、システム連携では、チャネルの柔軟性に問題、課題があることが、小項目での評価項目より確認することができた。これに基づき原因をさらに詳細検討し、具体的な対策案を企画できると考えている。

これらより、KLI インジケータ－項目の評点そのもの（1～5）についての議論よりは、各 KLI インジケータ－大項目で、相対的な比較を実施し、各行内および銀行間での強み、弱みの把握ならびにその原因の確認が有効であると考察される。

(3) KLI 評価インジケータ－説明改善点

インジケータ－ 30 項目中、8 項目では、評価レベル 5 の説明表現などについて、評価者から指摘事項をいただいた。特にツールの利用観点も含めて高度に実現できていることを評点 5 と表現した部分があるが、本質的に高度であることと、その方法論（ツールの利用）の観点が混在していた。また、他システム連携に関する項目では、一部 SOA や ESB という DSE バンキング・アーキテクチャーの直接の範疇ではないものが存在し、評価軸のぶれが生じる点あるいは不明点のご指摘をいただいた。これらの項目については、インジケータ－項目の分割と説明表現の改善が必要と判断し、修正を実施した。

(4) お客様評価コメントからの結果総評

評価者からいただいたコメントから今回の KLI 評価の

価値を以下のように評価した。

- ・評価者が、網羅性を持った評価項目でセルフ・アセスメント可能となり、継続的なレガシー・コントロール・ループを回す際の数値化した指標になりえる。
- ・担当者レベルよりは、全体的な管理者に継続的なご評価時の一助にお使いいただくことが有効である。
- ・評価は、評価者の主観的基準となり異なる場合がある。このばらつきを抑えるため、評価指標自体をさらに明確にする必要がある。
- ・同環境の評価者で評価が異なった点については、その項目での相違が何によって生じたのかを確認・検討することが重要であり、本質的に差異があるのであれば両者の認識共有や、気付きとすることにもつながる。
- ・評点そのものよりも、全体のバランス、強み、弱みや優先度などに応じた対応案の検討、策定が重要である。

4.4 評価結果の活用

まず、KLI 大項目については前述のとおり、相対的な比較を実施し、各行内および銀行間での強み、弱みの把握ならびにその原因の確認が可能となる。この把握結果から、組織として対策が必要となれば、対策案の検討、優先順位付けなどを実施し、組織的に目的とするレガシー・コントロール・ループを回し始めることが可能となる。KLI 中項目では、同環境、同レベルの評価者で評価が異なった点について、その項目での評価相違が何によって生じたのかを確認、検討することも有効である。例えば、フィールドテスト結果では、お客様と IBM の評価相違が確認されたが、両者の認識共有や改善活動の立案に活用でき、その実現によってお客様満足度の改善に結び付けられることを確認した。また、本質的に差異がある項目については、両者にとって有効な気付きとすることにもつながった。図4に示すように、アウトソーシングなどでサービスをご提供している場合、お客様管理者と IBM 管理者双方で当評価を実施することの有効性も確認された。

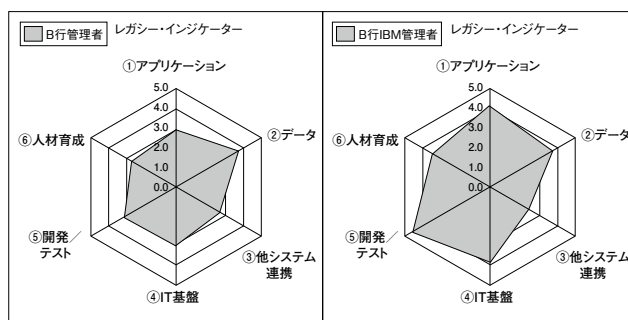


図 4. お客様と IBM 双方の管理者による KLI インジケータ－評価比較

5. 今後の展望

KLIをより効果的に活用するために、次の観点で改善、拡張を行うのがよいと考える。すでに本論の7地方銀行だけでなく、IBMの基幹系をお使いではないお客様からも引き合いをいただいている。

5.1 同業界でのKLIデータの蓄積とその活用

評価方法が確立すれば、次は、以下のような評価データの蓄積と活用法について検討する。

- ・地方銀行やメガバンクなどのお客様でKLIの評価を行っていただき、多数のデータを蓄積し、統計的手法も取り入れ、一定のベンチマークを可能とする。
- ・ある分野に対して相対的に強みを持っているお客様からその分野のベストプラクティスを集める。
- ・お客様へ有益なフィードバックをお届けできるようにする。

5.2 レガシー・インジケータの悪かった部分への改善案の明示

KLI本来の目的である、継続運用に必要な改善策の優先順位をお客様に提示するために、以下の対応を実施する。

- ・評価の結果お客様が課題と考えた分野に対して、適切なソリューションをマッピングして提示できるように準備する。
- ・中項目に対する評価だけでは、改善策が判断できないものに関して小項目を設定し、コンサルティングなどによる詳細分析を行う。アプリケーションやデータに関する確立した分析手法と連携できる体制を整える。

5.3 KLI自身のレガシー・コントロール・ループ

KLIによる評価メソッド自身が陳腐化しないために、以下のように改善作業を継続する。

- ・KLI評価の実施者よりコメントを必ずフィードバックしていただく仕組みとする。
- ・コメントに基づき、より適切な評価項目を追加や評価インジケータに対する説明文の改善を行う。
- ・新しいメトリクスやレガシー対策ソリューションを定期的にテクノロジー・スキャンしKLIに対してマッピングする。

6. おわりに

KLIは、シンプルな手法ながらお客様が潜在的に抱えている問題意識を可視化させ、レガシー・コントロール・

ループの検討を始める一定の観点として、大きな可能性があると考えている。貴重なコア・システムではあるがブラック・ボックス化しがちなシステムを、そのレガシー化度を把握しながら優先度をもって改善することにより、長く活用していただきたい。

また、金融以外の業種においても同様のアプローチが可能と考えられるので、本論文を参考にKLIの策定に活用していただければ幸いです。

謝辞

本論文執筆にあたっては、金融次期銀行基幹系アーキテクチャー検討タスクのメンバーやSIサービス提供部門の方々と技術的な議論をさせていただきました。ここにあらためて深謝いたします。

参考文献

- [1] Michael C. feathers: "Working Effectively with Legacy Code", Prentice Hall (2004).
- [2] IBM金融ソリューション・カタログ, http://www-06.ibm.com/industries/jp/finance/pdf/ibm_finance_solution_catalog.pdf
- [3] M M Lehman: "Laws of Software Evolution Revisited," (1997.01).
- [4] IEEE Std 982.1-1988, "IEEE Standard Dictionary of Measures to Produce Reliable Software," (1989).
- [5] 銀行基幹系システムアーキテクチャー白書2010, 日本アイ・ビー・エム株式会社, 金融インダストリー・ソリューション, (2010).



日本アイ・ビー・エム株式会社
テクニカルリーダーシップ
金融クライアントIT推進
シニアITアーキテクト

溝田 聡 Satoshi Mizota

[プロフィール]

1984年、日本IBM入社。金融のお客様の業務開発プロジェクトやシステム基盤構築を多数経験。現在は銀行系ソリューションの提案を中心に活動する。シニアITアーキテクト

MIZOTA@jp.ibm.com



日本アイ・ビー・エム株式会社
テクニカルリーダーシップ
金融クライアントIT推進
アソシエイトプロジェクトスペシャリスト

茂木 大地 Daichi Mogi

[プロフィール]

1992年、日本IBM入社。複数業種のお客様担当SEとして、基盤構築やシステム開発プロジェクトを経験。2000年以降、金融機関におけるオープン系開発プロジェクトおよびソリューション提案を中心に活動する。