

# バーゼルⅢ流動性規制導入案件の 対応事例

金融サービスビジネス部

河内 浩人



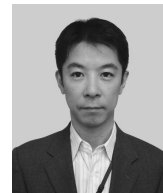
金融サービスビジネス部

稲富 祐介



金融サービスビジネス部

鈴木 拓也



## 1. はじめに

本稿では、国内金融機関のバーゼルⅢ流動性規制（流動性カバレッジ比率）導入案件の対応事例を報告する。

バーゼル規制への対応は、1990年代以降の銀行業務のシステム開発に携わったことがあれば、何らか関連を持った経験のある技術者は、当社にも多く在籍するものと思われる。

古くは「BIS規制」（バーゼルⅠ規制に該当）、「新BIS規制」（バーゼルⅡ規制に該当）という呼称の方が通り良く<sup>\*1</sup>、Ⅰ～Ⅱを通じて、銀行の自己資本比率の国際統一基準が定められ、日本を含む各国にて銀行規制として採用されたことにより、銀行のリスク管理業務において市場リスク管理、信用リスク管理、オペレーショナルリスク管理の高度化が促進された。バーゼルⅢでは2000年代後半のサブプライムローン問題、リーマンショックによる金融危機を受け、定量的な流動性規制（最低基準）を制定することが定められており、その一つが流動性カバレッジ比率（LCR：Liquidity Coverage Ratio）である。

## 2. バーゼルⅢ流動性カバレッジ比率規制の概要

### 2.1 流動性カバレッジ比率の目的

当局が示している流動性カバレッジ比率の指標/目的は以下のとおり。

「ストレス状況下でも市場から流動性を調達することができる高品質の流動資産（適格流動資産）を、短期間（30日間）の厳しいストレス状況下におけるネット資金流出額以上に保有することを求める」

これを読解するためには、以下①～③のキーワードの意味す

るところを理解する必要がある。

#### ① ストレス状況下

流動性リスクの顕在化した状況（個別行の財務悪化による資金繰り不調、金融市場の混乱による市場取引不全、これらが相互に作用し同時発生している状況）を指すが、具体的に、以下のとおり、想定状況が示されている。

- a. 信用格付けの3段階格下げ
- b. リテール預金の一部流出
- c. 無担保調達の停止
- d. 有担保調達の不全

（後述の適格流動性資産を担保とする調達のみ可能）

この状況下における定量的な流動性の評価が以下②③に定義される。

#### ② 高品質の流動資産（適格流動資産:HQLA）

ストレス状況下においても大きく減価することなしに換金できる資産であり、かつ換金にかかわる障害がないものを指す。

バーゼルⅢでは、これに算入できる流動性資産を細かく規定し、さらに品質度合いに応じてカテゴリズ/レベル付けをしており、レベルに応じて算入上限を掛目/パーセンテージ<sup>\*2</sup>で定めている。例えば、「現金」「中央銀行預金」などは100%算入可能なレベルにカテゴリズされるが、「社債」については格付けに応じて、50%～80%の算入上限幅を定めている。また金融機関が発行した社債は対象外であり、適格流動資産:HQLAとは見做されない。

\*1) バーゼル規制を制定するバーゼル銀行監督委員会の事務局が、国際決済銀行(BIS)にあることから、こう呼ばれた。バーゼル銀行監督委員会と国際決済銀行は別組織なので、「バーゼル規制」の方が呼称として正確であり、かつ現在では一般的である。

\*2) 掛目/パーセンテージについては、バーゼル銀行監督委員会による影響度調査の過程で、幾度か緩和の方向で改定されている。上記例示の掛目/パーセンテージについても中間案として告示されたものなので、ご留意いただきたい。

### ③ ストレス状況下におけるネット資金流出額

30日間のストレス状況下において、予測される資金流出額の累積から資金流入額の累積をマイナスした額面を指す。資金流出については、30日の間に満期を迎える負債項目に加え、バランスシート上の負債残高に対し、商品性/品質度合に応じ、流出率が掛目/パーセンテージ\*2として規定され、この範囲においては流出するものとして算入することが定められている。例えば、リテール預金においては預金保険制度による保護対象範囲や、給与振込口座として利用されている口座は“安定している”と見做され、流出率は7.5%、大口預金、富裕層預金、外貨預金、インターネット預金は相対的に“安定していない”と見做され、流出率は15%となる。一方、資金流入については、バランスシート上の資産項目で、かつ30日の間に償還を迎えるものが対象範囲となるが、ストレス状況下を前提に“健全な債権から生じる”と判定される予定分しか、算入することは認められない。

また、オフバランスについても、デリバティブ取引の担保に相当する額面の信用格付け格下げに伴う追徴担保の予測額(担保種類に応じた担保額面の掛目/パーセンテージ\*2により求める)や、流動性供給/信用供与ファシリティの未使用額の一部、若しくは全部分(被供与先の属性に応じた掛目/パーセンテージ\*2により求める)を資金流出に算入するものと定められている。

## 2.2 流動性カバレッジ比率の計算式と当局報告サイクル

図1 流動性カバレッジ比率計算式

$$\text{流動性カバレッジ比率} = \frac{\text{高品質の流動資産（適格流動資産：HQLA）}}{\text{（30日間の）ストレス状況下におけるネット資金流出額}} \geq 100\%$$

分母を「(30日間の)ストレス状況下におけるネット資金流出額」、分子を「高品質の流動資産(適格流動資産:HQLA)」とし、分母に対する分子のカバー率を求めるものである。つまり、30日間のストレス状況下で発生が予測される資金流出額に対しこれに充当可能な即時に資金化できる自前の資産を比率で示すものとなる。国内の各銀行/銀行グループにおいて、単体/連結、それぞれで流動性カバレッジ比率を100%以上とすることが求められる。正式には2015年より規制導入した後、段階的に2019年までに100%以上とすることが義務付けられる。当局への流動性カバレッジ比率の報告サイクルは月次であり、月末日を基準日とし、月末日以降30日間をストレス期間とした場合の流動性カバレッジ比率を算出し、基準日翌月の中旬\*3に報告するものとなっている。

\*3) 報告日は当初、毎月暦日14日目とされていたが、見直しされている。

## 3. 開発プロジェクトの背景と課題

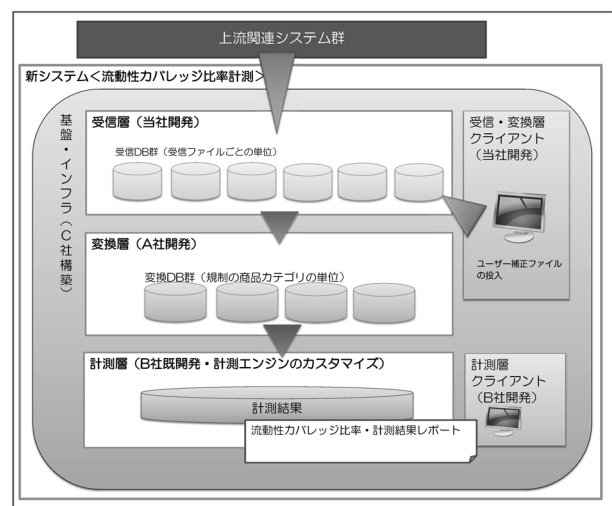
### 3.1 マルチベンダー開発と当社開発担当範囲

システム化にあたり、当初より前提となっていたのが、ある大手ベンダーが既に他の顧客向けに導入した実績のある流動性カバレッジ比率計測エンジンの採用で、取引データを規制で定められた分類と同様の基準で分類し、必要項目を抽出して入力すると、規制上のレベル付けや掛目/パーセンテージを反映し、流動性カバレッジ比率を算出するものである。

これのカスタマイズ導入と、既存システムより取引データ/取引明細を受信する機能と、収集した取引データを流動性カバレッジ比率計測エンジンのインプットに合わせて変換する機能を、開発領域として分類し、「受信層」、「変換層」、「計測層」の3層によるアプリ開発を行うこととなった。

当社が請け負うことになったのは「受信層」である。階層別に、それぞれ、別のベンダーが担当し、さらに全体の基盤部分を請け負うベンダーも参画し、計4ベンダーによるマルチベンダー開発を行うこととなった。

図2 システム階層図



### 3.2 「受信層」固有の課題

当社が担当した「受信層」の主要要件は、流動性カバレッジ比率算出の元データとなる取引データを既存の情報系システムや市場系システムから受信し、新システムの受信DBに格納し、これを「変換層」に連携すること、また、受信DBは極力、原取引データの形式を保全し、ユーザーが流動性カバレッジ比率の計測結果から辿って取引明細の確認ができるようにすることであった(取引属性や、取引先属性により規制に応じたカテゴライズやレベル付/掛目の決定を行なうため、勘定データのみでは足りない。取引の明細情報が必要となる)。この要件を実現するにあたり、計画当初からの設計の重要課

題となったポイントを、以下①②に要約する

#### ① 多量の受信ファイル/相手システム

受信対象とするデータは、リテール/法人、国内/海外の垣根無く、銀行の取引データほぼすべてであるが、これら取引データを保有する既存の情報系システム/市場系システムは、リテール/法人、国内/海外それぞれ個別に存在する。加えて基本設計工程開始段階では、規制の告示内容は未だ定まっておらず、後の変遷にも耐え得るよう、幅広く必要と考えられるデータを収集する必要が生じた。このため、関連システム/受信ファイルの数は膨大となり、かつ関連システムのプラットフォームに応じてファイル/データの形式も多様となった。限られた開発期間の中で、多種多様なインタフェースファイルを取り込む開発を行うためには、業務上はエンティティ(リテール/法人、国内/海外)や商品、相手システムごとに分類される受信ファイルを、データ形式、受信形式によってパターン化し、受信処理を共通化することで、開発効率を向上させることが必要不可欠となった。

#### ② 受信ファイルの到着監視と遅延監視

受信するデータは月末締めで確定したデータが対象となるが、各関連システムで月末データの確定タイミングはまちまちであるため、データ受領タイミングも応じて多岐にわたる。例えば、海外系システムのデータは確定データの受領まで期間を要するため、各海外拠点別に暫定データから確定データまでの数世代のデータを受領し、後続処理に要する時間を見極め、計測結果算出～当局報告までに間に合うデータを選択して引継ぐ必要があった(確定データが間に合わなければ、暫定データを使用し、遅滞なく計測結果を作成することが優先される)。またリテールのデータについては、データ量が膨大(預金や貸付の取引明細は数千万規模、同取引のキャッシュフローについては数億規模)となるので、当局報告リミットに抵触しないよう、到着後、即時に受信処理から「変換層」への連携を行うなど、受信ファイルの特性に応じた24時間フルタイムの到着監視と、何らかの要因で到着が遅延した際に、後続の処理リミットに応じてタイムリーにアラームを発信する遅延監視の具備が必須となった。多量の受信ファイルの対応同様、開発効率化を推進するため、到着監視、遅延監視とも、それぞれ、個別のファイルの受信処理と切り離し、パトロール機能として独立化することを当初より目論んだ。

### 3.3 マルチベンダー開発における課題

マルチベンダーによる開発が採用されるに至った主な理由は、開発規模に比し開発期間にあまり余裕が無いことに関するリスクの軽減である。開発規模は、リテール/法人、国内/海外の垣根無く銀行のほぼすべての取引データの取込を行うという点で前例が少ないこと、取引データの収集が一元的に可

能となる統合された情報基盤は整備～開発中のため、その利用が見込めないことより増大し、関連システムも含めたトータルの見積りは数千人月規模となった。対して開発期間は、制度案件としてリリース時期は定められており、規模見積りに応じて余裕を持たすことは不可であった。

マルチベンダー化により、各アプリ領域の並行開発を行うことで、1ベンダーあたりが抛ししなければならないパワーは限定的となり、シンプルな工数/期間の判断のみで言えば、各ベンダー単体の開発計画は(大きくは)無理があるものでは無くなったが、ロス無く並行開発を実現するためには、新システム内のアプリ開発領域間のインタフェースを疎結合とすることが必要不可欠となる点、そのための設計/調整に要するパワーをコントロールし、トータルでマルチベンダー化のメリットを出していかなければならない点は課題となった。「受信層」において、最も重要だったのは当然のことながら、直下の階層である「変換層」とのインタフェースである。到着から受信処理終了までの受信ファイルごとのステータス管理と、規制上の業務/商品定義単位に構築される変換層処理群を起動させる条件/ロジック(商品の切り口で必要ファイルが揃うこと)を分離・分掌することにより、疎結合の実現を図った。

## 4. 開発課題への対応

### 4.1 多量の受信ファイルの実態

受信ファイルの送信元となる関連システムは、リテール系19システム、法人系11システムの計30システム。また、システムから取得できないデータ(システム管理されていない取引データ、若しくは必要な情報がシステム管理されていない取引データ)は、ユーザーが作成する補正データとして取り込む。受信するファイルの種類としては、システムインタフェースファイルとして153種類、ユーザー補正ファイルは31種類。関連システムがホスト系システムの場合、マルチレコードフォーマットのファイルもあり、受信ファイルを格納する受信DB/テーブル数としては203種類に及んだ。

#### ① 様々な特性のファイル

多岐にわたる関連システムから受領することや、関連システムに既存のファイル連携の機能がある場合はそれが流用されることもあり、受信するファイルの特性は多様である。以下に列挙してみる。

##### a. 文字コード

EBCDIC/EUC/Shift\_JIS/UTF-8などがある。当システムのDBMSはORACLEで文字コードはUTF-8であるため最終的にはUTF-8にコード変換してテーブルに格納する。

##### b. レコード形式

ファイルごとに固定長/可変長のいずれかの形式である。

c. 圧縮ファイル

圧縮せずに連携されるファイル、圧縮して連携されるファイルの両方のパターンがあり、圧縮ファイルは解凍すると一種類のファイルの場合と複数種類の場合がある。

d. 分割ファイル

データ容量が大きいファイルはデータ伝送の仕組みの制約や、そもそも関連システムでも分割して保有・管理されているなどの理由で、分割されて受け渡しが行われる。当システムでは受信後にファイルを結合して取り込むパターン、大容量の場合は受信する分割ファイル数分の取込を行うパターンに分けて対応している。

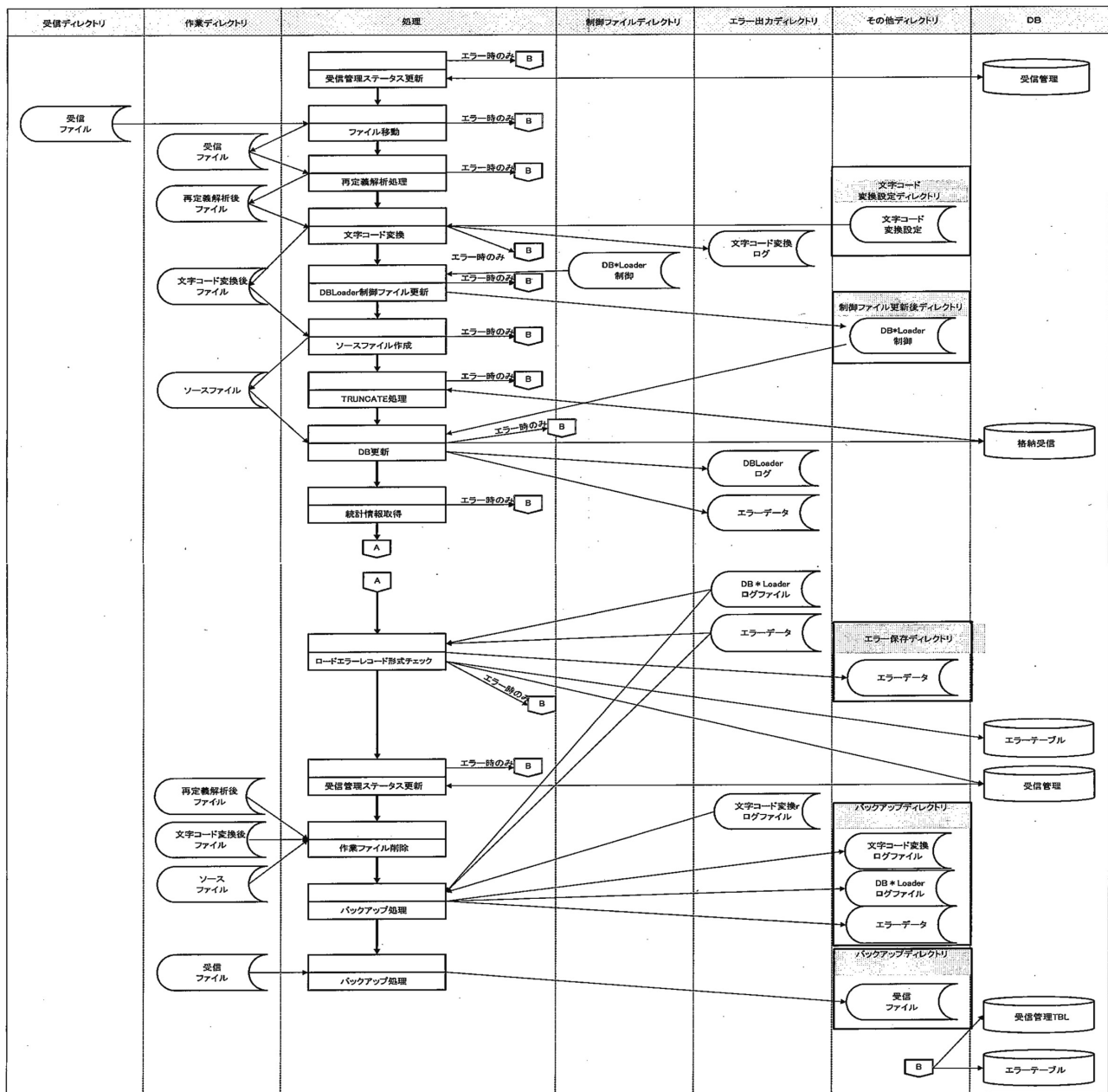
② 受信タイミング・頻度

当システムは現状月次バッチシステムであり、受信ファイルも月次での受け渡しがメインだが、関連システムの都合上、月次や週次で受け渡しを行うものもある。また、速報版/確報版など、複数回にわたって受け渡しが行われて都度洗い替えるファイルもある。

③ マルチレコードフォーマット

ヘッダレコードやトレーレコードを保有する、いわゆるマルチレコードフォーマットのファイルもあれば、明細レコードのみのファイルもある。また、明細レコードの種類が複数になるものもある。明細レコードのみ取り出してテーブルに格納するが、明

図3 受信処理パターン例



細レコードが複数種類になる場合は、明細の種類別に分解しておのおの対応するテーブルに格納する。

#### ④ ユーザー補正ファイル

ユーザーがマニュアルで作成し、当システムに補正入力するものだが、1種類のファイルでも複数の物理ファイルに分けて作成されることもあり、その数は一定ではない。したがってファイル名も部分的に一定ではなくなってしまう。

### 4.2 受信処理の効率化・高品質化のための施策

効率的な開発の具体的な施策を以下に記す。効率性を重視し、処理を共通化しシンプルにパターン化することが、堅牢性/高品質化の確保にもつながっている。

#### ① 共通化

関連システムからファイルを受信して当システムのテーブルに格納するまでの処理の流れで、様々な特性のファイルの処理で必要となる機能を共通機能として切り出し、モジュール化した。開発の効率化と品質の安定化を狙ったものである。シンプルな機能で汎用性の高いモジュールを作成し処理フローとして共通機能を連携して使用する設計とすることで狙いを達成した。本当に特定のファイルに依存して制作が必要な機能以外はすべて共通化した機能だけを使って一連の取込処理を行っている。

#### ② パターン化

さらに一連の処理の流れが同じになるファイルについては、関連システムの垣根も外してグルーピング化して捉えることで取込処理を24のパターンに分類することができた。このことにより各パターン単位に詳細設計から結合テストまでの作業が可能になり、設計・制作・テストの効率化・品質の安定化が図れた。

#### ③ パラメータのファイル化

各共通機能はパラメータを受け渡すことで個別のファイルの差異に対応するが、ジョブ実行用のshellの引数としてそのまま渡すように作りこむとジョブの設定作業量が膨大になってしまう。そのためパラメータ自体をファイル化してまとめてしまい、ジョブ実行時に実行中の機能自身が必要なパラメータ値をパラメータファイルから取得するようにした。パラメータファイルの内容は、詳細設計書からExcelマクロを使用して半自動的に作成するようにし、作業の効率化と不具合の混入を防ぐようにもした。

#### ④ 共通化・パターン化によるその他の効果

文字コード変換にはホスト系システムにも対応している市販のツールを使用した。現場で使えるようになるタイミング

が遅かった。また、ホスト系の関連システムから受信するファイルの詳細なレイアウト仕様の提示が一部大幅に遅延した。しかし、機能を切り出しモジュール化して処理フローを構築したことで、外部要因による作業遅延の影響を極小化することが可能となった。

### 4.3 到着監視と遅延監視

システム/ファイルを超えた共通の仕組みとして到着管理があり、ファイルの形式・受信サイクルが異なっても、管理の仕組みは共通となっている。

管理の情報は、データベースの中のテーブル「受信管理テーブル」に一元的に記録され、様々な機能のプラットフォームとして利用される。

#### ① 到着管理機能

毎月、特定の期限までにレポートを作成することを要件としたシステムであり、情報元となる関連システムからのIFファイルの到着状況を管理する。

IFファイルの到着サイクルとしては日次・週次・月次とあり、月次には速報版・確報版と同じファイルを複数回受信するなど様々なケースがある。

受信予定の情報をもとに営業日カレンダーと合わせて、月次で次月の実際の受信予定日時を算出し、すべてのファイルの到着状況を管理する。

カレンダーは国内だけではなく、海外拠点のIFファイルには現地の営業日カレンダーを利用している。異なる営業日や時差も考慮された仕組みとなっている。

すべてのIFファイルは、受信前には「未登録」というステータスになっており、受信後に「仮登録」、データベースへの取込が完了した「登録済」となる。

受信処理内のファイル形式チェック機能によりデータ形式に異常が見つければ「エラー」というステータスになる。

これらの状況は、ユーザーが画面から確認ができるようになっている。

関連システムから受信するファイルだけでなく、ユーザーが毎月、補正入力するファイルもあり、それらも同様に管理されている。

#### ② 遅延監視

到着予定日時を既定の時間過ぎると、遅延として検知し、運用部門の監視端末に表示し、開発部門へ連絡が来る仕組みである。

到着予定時刻と遅延を検知するまでの猶予時間はファイルごとに個別に設定が可能となっている。

遅延を検知するべき時間は、後続の処理に遅延を生じさせる時間からであり、関連システムから送信されてくる到着予定時刻とは異なる。

このため、到着予定時刻と遅延を検知するまでの猶予時間はファイルごとに異なる設定ができるようになっている。

ユーザーが補正入力することになっているファイルも管理対象であり、期限までに補正入力されない場合は、開発部門からユーザー部門へ補正入力を催促することになる。

#### 4.4 「受信層」と「変換層」の連携

「変換層」の変換処理の実行可否の判断も「受信管理テーブル」を利用する。必要なファイルがすべて揃った段階で、変換処理を起動するためのトリガーファイルを作成する。

同じファイルを複数回受信する場合などは、特定の期限が過ぎた場合には1回でも受信していれば後続を起動してよい場合など、特殊な条件にも対応する。

ポイントは、トリガーファイルを作成する機能は「受信層」所管機能だが、変換処理の起動条件/トリガーファイルを作成する条件とロジックはパラメータ化し、「変換層」所管のテーブルに保有させた点である。これにより、「変換層」の業務要件の変更(今後も一定の頻度で発生することが予測される)による、インタフェース機能の影響を極小化し、双方の開発作業の干渉範囲を狭めている。

## 5. おわりに

当社が開発を担当した「受信層」の開発要件自体はシンプルであったが、制作しなければならない機能の物量と、調整しなければならないステークホルダーの数の多さは特殊であり、限られた開発期間も相まってプロジェクトの計画段階

で、QCDで言えば納期重視、そのために設計においては効率化/共通化を推し進めることが最も重要であり、優先すべきことと定義することに迷いは無かった。ただし、当然のことながら、掛け声ばかりで実現できるものでもない。幸い、当プロジェクトは、3次オンの頃からの大規模開発の経験者で、効率化を実際に具現できるノウハウを持った要員、情報系ミドルの開発経験が豊富で本案件と同様の課題を実際に解決してきた要員を、設計のリーダークラスとしてアサインできたことが、非常に大きかった。

処理の抽象化、部品化を上流設計工程で推進し、開発ピーク時に、どれだけ製造単位をシンプルに分割し、高生産性を実現するかというノウハウの活用は、かつて当社の金融開発部門でも得意とするところで、あまりにもドラスティックなやり方により、顧客によっては理解してもらえない事例もあったと記憶する。現在もちろん、エンジニアリングとして重要な要素と捉えられているものと思うが、実施のノウハウの継承という点では、リライト的な/完全新規ではない案件では難しい面もあり、本案件への参画は貴重な機会であったと考える。また、ピーク時の開発体制/期間を念頭/前提にした設計の実現に対し、上流工程期間中に都度、チェックポイントを持つことは、一定規模以上の新規開発の推進において、極めて重要であることを改めて認識する機会ともなった。これは同時に堅牢性、拡張性といった観点での品質を見極める要素ともなり得る。何よりも、設計の巧さ/拙さが、その後の工程における生産性/開発要員の適正な稼働にまで直結し、案件の成否を左右するという意識が浸透すること、技術者のベーシックな素養を形成するための意識として継承されていくことが、今後においても重要であると考えている。