

1

B to Bシステム実現に向けての諸考察: ビジネスACIDほか

Thoughts on the realization of the "B to B" system: Business ACID, etc.



日本アイビーエム中国ソリューション株式会社
技術理事(DE)
IBMアカデミー・会員
シニア・コンサルティングITアーキテクト

福場 芳正

Yoshimasa Fukuba

Certified IT Architect
Distinguished Engineer
Member of the IBM Academy of Technology
IBM Global Services Japan Chugoku Solutions Company

インターネットの本格普及に伴い、B to Bの実現に向けて数々の試みがなされ、最近それを支える多くの重要な基礎技術が提供されつつあります。

このような状況において、B to Bシステムにおける特性と必要条件について考えます。基本はあくまで取引(トランザクション)を完遂することですが、そのトランザクションは、従来のトランザクションと実行環境も必要条件も大きく異なります。それをビジネス・トランザクションと称し、それに必要な要件をビジネスACIDと定義して、その実現性可否を論じます。

さらにその基本の上に、ダイナミックe-businessに代表されるB to Bシステムに期待される諸要素に対して、最新の基礎技術がいかにかたえられるかを述べます。

今や「夢」であったダイナミックe-businessは、実現に向かって踏み出す時期に達しつつあります。しかも、それは多大なメリットをわれわれに提供するでしょう。

Various efforts have been made to realize "B to B" accompanying the rapid diffusion of the Internet, and a wide variety of important basic technology to support these efforts has recently been made available. Under these conditions, we need to think about the features and essential conditions of "B to B" systems. We are dealing here essentially with the execution of transactions, but we need to recognize that the implementation environment and the essential conditions for these transactions are completely different from those of previous transactions. These are referred to as business transactions, the essential conditions are defined as business ACID, and we will give consideration to whether or not their realization is possible. In addition, on this basis we refer to how the most recent basic technology is able to respond to the various elements expected of "B to B" systems as typified by dynamic e-business. We are gradually entering an era when the "dream" of dynamic e-business is at long last going to be realized. This is sure to bring us enormous benefits.

1.はじめに

B to B (Business to Business)という言葉は、広くは「企業間の電子商取引」を意味しますが、本論文では、インターネットを介してのB to Bについて論じます。なぜなら、インターネットはB to Bにおける今後の飛躍的発展の源泉となるネットワーク基盤だからです。一般的にも、インターネットを前提としてB to Bは議論されています。

ところで、その観点からのB to Bと、従来からも行われているEDIなどの電子商取引の異なる点は何でしょうか。それは、従来の専用線による「閉じた」ネットワークではなく、インターネットという「オープン」なネットワークを介することにより、事実上世の中のほとんどの企業、ならびにそれらの企業が持つ関連システムとの接続が可能になるということです。それは、ネットワークが世界に張り巡らされた交通網や電話網よりも、さらに高度な企業間の連携インフラストラクチャーを提供し、企業間のコラボレーションの飛躍的な向上と促進を可能にすることを意味します。

つまり、このシステム・インフラストラクチャーを活用すれば、新規顧客の開拓や、分断された非効率な従来の決済処理なども、すべて一連のプロセスとして電子的に処理できるようになります。そうした特徴により、経済面ならびに競争面において著しい利益を生み出すことが期待されています。現に、B to Bの早期実現に向けて多くの企業が対応し始めていて、先行者利益の享受を目指しているところもあります。

本論文では、B to Bの実現可能性を体系的な種々の観点から論じます。

2. B to Bのシステムの特長

B to Bの体系的な特長は、大きく分けて本質の課題である企業間の商取引(トランザクション)を電子的に行う点、必然的に必要となってくる多様な企業間を連携する点、さらにその構築における特長にあります。

2.1. ビジネス・トランザクション

1章でも述べているように、B to Bはビジネスにおける一連の処理をネットワーク上で行おうとするものです。これまで、商取引の一部はネットワーク上で行われてはいました。しかし、それは体系的にはトランザクション処理といわれる作業単位(UOW:Unit Of Work)で単発的なものでした。例えば、発注処理などがそれです。

B to Bにおけるトランザクションは、そのビジネス・プロセス

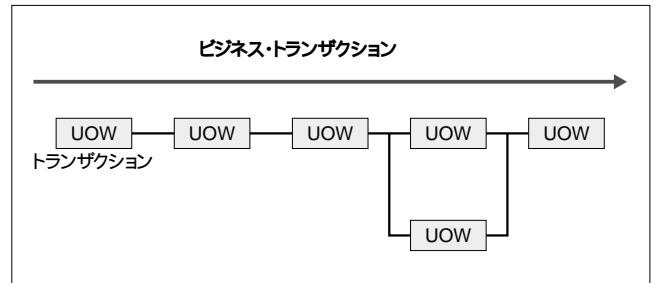


図1. ビジネス・トランザクション

に応じて従来のトランザクション処理が複数存在することによって構成されます。このようなトランザクションを、本論文では「ビジネス・トランザクション」と呼びます。図1にその関係を示します。

ビジネス・トランザクションは、企業間をまたがった多くのプロセスを経るため、終了までに長い時間が必要です。つまり、ロング・トランザクションなのです。企業間をまたがるということは、通常はシステム間もまたがることを意味します。さらに、通常のビジネス・プロセスのように、分岐して並行処理も行われます。

プロセスの遂行過程において、やむを得ずビジネスの取り消しも起こり得ます。それも、一部の取り消しであったり、全部の取り消しであったりします。「取り消し」とは、システム的にはその処理がなかった状態にすることです。また、システムのトラブルによって、プロセスの遂行の途中で停止せざるを得ない場合もあります。特に、インターネットという堅牢^{けんろう}とは言い難いシステムを介しているため、その確率も残念ながら高くなります。この場合に必要なのは、ある時点までの復帰とその後の回復処理です。

従来のトランザクション処理レベルでは、復帰や回復は、ほとんどの場合システム(ここではOLTPやDBMSを指す)によって自動的に行われていました。つまり、取り消しならデータベースのレコードの絶対値がそのトランザクション実行以前のもので等しくならなければなりません。しかし、ビジネス・トランザクションにおいては、ビジネス的な意味での取り消しや復帰・回復の方が、より重要かつ実質的な意味があります[参考文献1]つまり、発注なら取り消しが発生したときに発注量の絶対値が前の値に戻る必要はなく、取り消しに伴って消滅した発注量が差し引かれることが重要なのです。このような処理を、コンペンセーション(Compensation)といえます。

2.2. システムの多様性とダイナミズム

次の特長は、新規の顧客を含めた多様な企業との、ネットワークを介したビジネスの遂行です。従来のように同一企業系列や長年ビジネス取引を行っていた企業間のみではなく、新規の企業との取引がダイナミックに行われます。そこに、ビジネスの大きな飛躍の可能性が秘められています。しかし、多様な企業を相手

にすると、相互接続に困難が生じます。なぜなら、システム的には相互接続の各レイヤーにおけるプロトコルが統一された環境ではないからです。

現在では、ネットワーク・レベルではインターネットでの接続、つまりIP(Internet Protocol)での接続によってほぼ統一されたといえますし、それがB to Bへの期待と拡大の源となっています。しかし、それより上のレイヤーのプロトコルは、各企業によって種々さまざまです。それには、レイヤー的に見るとHTTPのようなアプリケーション基盤的なものから、よりアプリケーションに密着したビジネス契約的なものまであります。つまり、納期回答要求において3労働日以内に回答がなければ、そのビジネスを破棄するなどです。しかも、それらの企業(システム)はネットワーク上に刻々と現れては消えていくのがB to Bの特徴です。B to Bシステムがそのダイナミズムに追随する能力を持っていないと、前述のビジネス飛躍達成の可能性を失います。

さらに、多様な企業間で連携を行うほど、相手の企業の窓口が必ずしもオープンしている保証はできなくなりますが、それにもかかわらずビジネスは遂行する必要があります。また、予想しない相手からのリクエストもあり得ます。これらは、結合の形態として疎な側面の要求を意味します。

2.3. 迅速かつ柔軟なシステム構築

現在のビジネス環境を最もよく表現している言葉は、「変化の時代」でしょう。その変化に迅速に対応しない企業は、利益獲得の機会を逃すだけでなく、次々と脱落します。その変化の促進に重要な役割を担っているのもインターネットなのです。インターネットによるグローバルな変化は一層促進され、多様な企業とのビジネスの機会を提供します。しかし、それは同時にシステムの形態もビジネスのやり方も多様であることを意味します。だからといって、そのような環境において、柔軟かつ迅速に

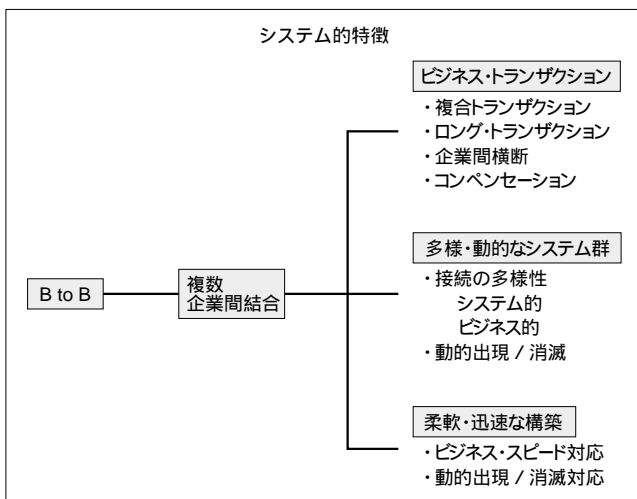


図2. B to Bシステムの特徴

対応しない場合は、インターネットによって提供されるビジネス機会の拡大をみすみす見逃すこととなります。

図2に、その特徴をまとめます。

3. B to Bに必要なシステム要件

2章の内容を踏まえて、B to Bに必要なシステム要件について考えていきます。

3.1. ビジネス・トランザクションのシステム要件

この問題の前に、従来のトランザクション・システムにおける要件を考えてみます。この場合に重要なのは、トランザクションがいかにシステム的に間違いなく確実に処理できるかでした。トランザクション処理の完全性を満たす要件は、ACID(Atomicity, Consistency, Isolation, and Durability)と呼ばれています。

トランザクションの終了時に、「処理すべてが完了している」が「まったく行われていない」のうち、どちらかを保証するのがAtomicity(原子性)であり、終了状態がどうであれデータベースの一貫性が保たれている要件がConsistency(一貫性)です。さらに、複数のトランザクションを実行したときに、ほかのトランザクションから影響を受けずに実行できるのがIsolation(独立性)です。そして、トランザクションの処理結果が永続的に保証されるのがDurability(永続性)です。

Consistencyは、ユーザー・プログラミング上での考慮が必要ですが、その他は現在においてはOLTP(Online Transaction Processing: オンライン・トランザクション処理)などの製品によりシステム・レベルで保証されています。

しかし、B to Bシステムの対象は、それらの個々の処理をまとめた一連のビジネス・プロセスです。つまり、前者のシステムが個々のトランザクション単位でACIDが保証されればよいのに対して、B to Bでは個々のトランザクションの集まりであるビジネス・トランザクション単位で保証されることが求められます。

この場合のACIDとは、何でしょうか。それは、ビジネス・トランザクションが最後まで正常に処理された場合は当然ですが、そうでない場合でもビジネス的な意味で従来のACIDを保証することです。端的にいえば、従来の正常または異常処理をも含んだ企業間取引が、B to Bによるビジネス・トランザクション上でも同等に行われ、ビジネス契約上問題が発生しないことであると筆者は考えます(5.1.2節を参照)。そこで、本論文では、後者のACIDを「ビジネスACID」と呼びます。

ビジネスACIDの保証は、システム的ではなくアプリケーション的に行われる必要がありますし、その方が事実上の意味があり

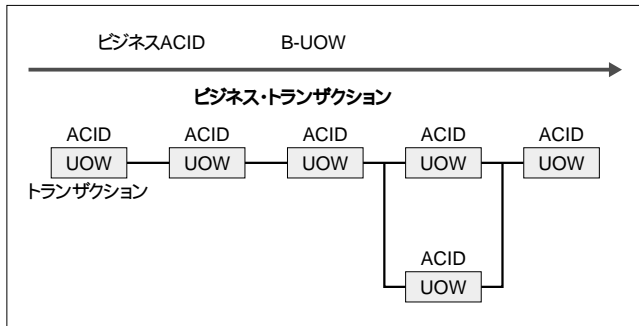


図3 . ビジネスACID、B-UOW

ます。このような対応は、「コンベンション」と呼ばれ、システムに対して行われた行為を、システム的な意味合いではなくアプリケーション的な意味合いで戻す、または矛盾のないようにすることです。これには、当然ビジネス契約上の補償も含まれます。

2.1節でも触れたように、ビジネス・トランザクションはロング・トランザクションです。この一連のビジネス的に意味のある仕事の単位を、本論文ではB-UOW(Business Unit Of Work)と呼ぶことにします。図3に、以上の関連をまとめます。

従来の発想だと、ACIDの保証として、B-UOWの間に必要なリソースにシステムのロックをかけておきます。そうすると、ほかのプロセスの処理の遅延を招くだけでなく、処理の時間的長さ故にデッドロック発生の確率も高くなります。そのような理由により、ビジネス・トランザクションのようなロング・トランザクションでは、システム的にロックをかけるのは断念せざるを得ない場合が多くあります。そのため、従来のトランザクション処理の概念からすると、UOWの間にロックをかけることによってACIDの保証を行うところを、コンベンションで行わざるを得ません[参考文献2]

さらに、2.2節でも述べたように、結ばれるシステムは多様であり、システムをまたがったトランザクション性は保証されていない場合が実際には多いので、B-UOW間をシステム的にUOWとして保証することは困難です。仮に、システム間のトランザクション性が保証されていたとしても、現実にはそれらを結ぶインターネットの信頼性は高いとはいえません。また、ネットワーク・トラブルによってロックが保持されたまま長時間放置される可能性もあります。ここにも、システム的にロックをかけるのは断念せざるを得ない場合が多くあります。

こうした複数の理由により、B to Bシステムではコンベンションの仕組みが必須といえます。

コンベンションを行うには、システムがACIDを保証するのにシステム・ログを必要とすると同様に、アプリケーション的のどのような処理を行ったのかのアプリケーション・ログが必要となります。さらに、そのログを入力してアプリケーション的に戻す処理ロジック、いわゆるコンベンション・トランザクションが必要です。

3.2. システムの多様性とダイナミズムにこたえるシステム要件

2.2節で触れた「システムの多様性」は、二つの大きな側面を持ちます。一つはシステム・インフラストラクチャー的な側面であり、もう一つはビジネス的な側面です。しかし、この多様性に影響されて、システム構築が制約を受けるのは可能な限り避けなければなりません。システム・インフラストラクチャー的な側面においては、前述のようにIPよりも上のレイヤーが多様なプロトコルやインターフェースで成り立っているのが事実です。

同様に、ビジネス的な側面では、提供されるサービスの内部処理もさまざまです。しかし、重要なのは、内部処理の仕方よりも、サービスの提供に当たっての外部に対するインターフェースです。さらに、サービスの提供時間やキャンセル時の取り消し料などのビジネス・ルールも当然重要であり、それなくしては現実のB to Bは成り立ち得ません。よって、このような多様かつ幅広い接続情報を必要とするシステム間を連携してB to Bシステムを構築するには、こうした情報を明確に宣言し、他企業(システム)からその情報が容易に取り込め、さらに取り込んだ情報によってシステム的に相互に接続されなければなりません。

しかし、B to Bは動的なシステムであり、真に求められるのは、こうした多様なシステムのダイナミックな結合を提供する手段です。そのためには、前述の情報がネットワーク上にダイナミックに公開されて利用できることが必要となります。また、その利用も何らかの共通化されたインターフェースを介して利用できなければなりません。そのようにすることにより、サービスやプロトコルから独立したシステムの構築が可能になります。

また、相互のシステム環境、運用環境が大きく異なるので、疎結合にこたえられる手段が必要となります。

3.3. 迅速かつ柔軟なシステム構築のための要件

2.3節の「迅速かつ柔軟なシステム構築」に関していえば、B to Bの環境で構築すべきシステムの形態は、「つくる」というよりは、既にネットワーク上に存在する多様なシステムとコラボレーションすることです。その世界では、下位レイヤーのIPを除いては、プラットフォームやプロトコル、言語、さらには交換されるデータの相違をも超えて、機能(サービス)を主体として柔軟かつ迅速に結合できることであり、3.2節の要件に大きく依存します。

このことは、別の見方をすればコンポーネント指向への大きな傾斜を意味しています。あるレイヤーから下は、何らかの方法で隠ぺいされた機能(サービス)のインターフェースのみで認識される必要があります。現実のシステムの世界でこの実現を困難にしているのは、ビジネス・ロジックとビジネス手続き(コントロール)が一つのトランザクションの中で混然一体と

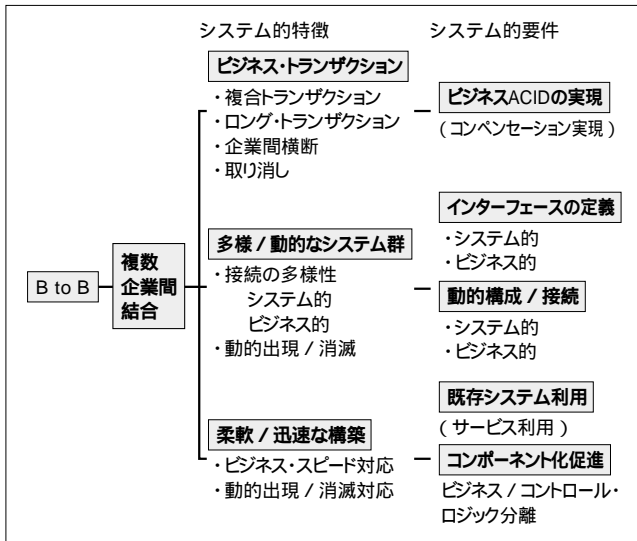


図4. B to Bに必要なシステム要件

なっている場合が多いことです。

今後は、個々のトランザクションの処理ロジック、つまりビジネス・ロジックとそれらを組み合わせる一つのビジネス単位にするビジネス・コントロール系を分離して考え、設計・製作することです。そうすることにより、部品の活用が容易になるとともに、処理ロジックのコンポーネント化も促進されます。また、ビジネスのプロセスが変わった場合や新しい企業やサービスを組み込む場合も、コントロール・ロジックのみに手を入れるだけで済みます。

こうした関係についてまとめたものが、図4です。

4. B to Bを取り巻くシステムの現状

4.1. 多様なシステム間の接続と稼働

4.1.1. ネットワーク・レイヤーの接続

前節の、「多様なシステム間をダイナミックかつ迅速につなぐ」のパフォーマンスに関しては、基幹網は高速化のための技術もWDM(Wavelength Division Multiplexing)などで実現されつつあり、着実に強化されつつあります。一番の問題であったラスト・ワンマイルのパフォーマンス・ネックも、既存の電話回線を使用して数Mbpsの伝送速度を実現するADSL(Asymmetric Digital Subscriber Line)の急速な普及や、末端での光ファイバー網の拡大などで、解決されつつあります。その脆弱性の飛躍的改善は図られていませんが、コンベンションに代表されるように、それを認識した上での対応が図られつつあります。

4.1.2. ネットワークより上位層の接続と稼働

IPより上位の層では、システムの統一によって企業間を接続

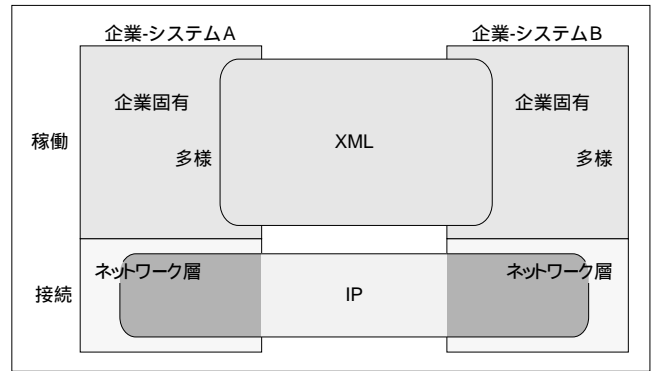


図5. 多様なシステム間の接続と稼働

するというアプローチは現実的ではありません。個々の企業が、多くの既存のシステムを抱えているからです。また、これより上のレイヤーは単に接続されるだけではなく、その接続を前提としてアプリケーションがビジネス的なやり取りをするわけなので、表現方法も「稼働」とした方が適切です。

ここでの現実解は、そのシステムの多様性にもかかわらず、接続して稼働することが可能な基盤が提供されることです。しかし、考えてみるとB to Bにおけるデータは単純なデータではありません。例えば、ビジネス・ルール、ERP、CADなどのデータが存在するので、これらを柔軟につなぐ必要があります。つまり、複雑な構造化と、アプリケーションやシステムから独立した自己記述可能なデータ形式の表現が可能でなければなりません。この要求にこたえるものが、XML(eXtensible Markup Language)です[参考文献4]。多くのB to B関連製品が、XMLを使用することによって多様なシステム間を連携しようとしています。

図5に、以上の接続と稼働についてまとめます。

4.2. Web ServicesとSOA(Service Oriented Architecture)

さらに注目すべきなのは、単に静的に接続する点だけでなく、3.2節で述べたB to Bの特色である動的な接続を満足する基礎技術の開発と普及が急速に進みつつあるという点です。それを実現しているのは、SOAP(Simple Object Access Protocol)、UDDI(Universal Description, Discovery, and Integration)およびWSDL(Web Services Description Language)です。これらはいずれもXMLをベースとしています。

SOAPによってオープンなメッセージの伝達の基礎が提供され、それを利用してUDDIでビジネス・サービスの登録と検索のためのリポジトリが作られ、WSDLによってそこに登録するビジネス・サービスのインターフェースが定義できます。この3者の組み合わせにより、企業はインターネット上で自己のビジネスに必要な他企業の「公開」されたサービスを「検索」し、「結合(バインド)」してEnd-to-Endで利用するということがダイナミックにできるようになります。もちろん、反対に自分の持っているサービス

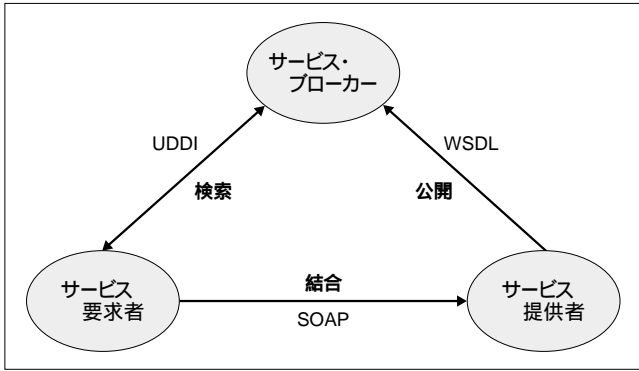


図6 . SOAP、UDDI、WSDLの関係

を登録することによって、新規ビジネスの開拓も可能です。
 このような基礎技術は、ビジネス・プロセスの改善と変革に大きく寄与することが予想されます。詳しくは、参考文献5を参照してください。

図6に、SOAP、UDDI、WSDLの関係を示します。

このように、ネットワーク上に公表・配置されて起動することが可能なサービスは「Webサービス」といわれ、それを支えるアーキテクチャが「SOA(Service Oriented Architecture)」です[参考文献5]つまり、ここでいう「サービス指向」は、サービスの自動検出と使用をサポートするための記述方法や編成方法に重点を置くアーキテクチャです。その核となるUDDIは、2000年9月に発表され[参考文献6]XML、HTTP、DNSといった国際的な標準化団体のW3CとIEFTの標準の仕様です。さらに、SOAPの使用により、多様なプラットフォーム間での接続と稼働性を実現しています。

しかし、ここで冷静に考えなければなりません。確かに、企業間がダイナミックにつながるというレベルの問題は、Webサービスの出現により大きく進展してきたように見えますが、現実にはそうでしょうか。また、B to Bがシステム的に最終的に実現したいことは、ビジネス・トランザクションの確実な実行でないと意味がありません。また、その迅速かつ柔軟な構築も重要です。

5. ビジネス・トランザクション実現の可能性検討

ビジネス・トランザクションの始めから終わりまでは、決して単純かつ平安なものではありません。前述のように、それは従来のようなシステムによってACIDが保証された単一システム内の1回の実行ではなく、多様なシステム間をまたがっての複数のトランザクションの実行です。しかも、信頼性が高いとは言いがたいインターネットを介して行われなければなりません。

ここでは、ビジネス・トランザクションの実現における完全性(3.1節で述べたビジネスACIDの実現が可能であるかという点)

と、それが保証されるとしても、それらの個々のトランザクションを多様なシステム間をまたがって一つのビジネス・プロセスとして完成することが可能であるかどうかについて検討します。

5.1. ビジネスACIDは成立するか

5.1.1. コンペーンセーションの実現可能性

トランザクションは、システム的にはACIDを保証することを目標としています。同様に、ビジネス・トランザクションにもビジネス的なACIDが求められます。

前述のように、ビジネス・トランザクションにおいては、そのACIDの保証に関して、もはや従来のシステムによるトランザクションACIDにすべてを期待するのは無理で、アプリケーション・レベルでの対応が必要となります。そのための手段の重要な要素の一つが、コンペーンセーションです。それが可能でないと、ビジネスACID自体が「絵に描いた餅」になります。

コンペーンセーションの検討のために、通常のビジネス行為をモデルとして考えてみます。ここでのトランザクション(取引)のビジネス行為は、買い手は権利(商品)の獲得とその対価の支払いであり、売り手は権利の譲渡とその対価の受け取りです。

このモデルにおけるコンペーンセーションの実現性を考えると、困難さは2点ほど考えられます。一つはコンペーンセーションのシステム的なロジックの複雑さであり、もう一つはそれが現実のビジネスにおいて可能かどうかです。

まず、コンペーンセーション・ロジックについて検討します。これは、売り手が権利(商品)の復活と対価の払い戻しであり、買い手が権利(商品)の消失と対価の受け取りです。システム的な実現手段は、両項目の増減のみなので複雑ではありません。表1にそれを示します。

現実のビジネスにおける可能性については、前述の増減に関して考えると増えたものは手を付けられない限り、減らすのは簡単です。しかし、一度失ったものを取り戻すのは困難な場合が多くあります。

これを前述のモデルに当てはめると、権利(商品)の獲得のコンペーンセーションはその放棄です。このケースの場合は、ビジネス・プロセスの途中で

表1 . コンペーンセーション

通常取引

| | 売り手 | 買い手 |
|--------|-----|-----|
| 権利(商品) | - | + |
| 対価 | + | - |

コンペーンセーション

↓

| | 売り手 | 買い手 |
|--------|-----|-----|
| 権利(商品) | + | - |
| 対価 | - | + |

あり、最終契約に至っていないが故に消費(手を付ける)は行われずにそのまま残っています。もちろん、そのためには契約成立まで手を付けられないようなシステムの仕組

みを考慮する必要があります。また、対価の支払いのコンペーションは、対価の払い戻しを受けることであり、相手が既に使用していたら困難な状態になります。しかし、この支払いは一般にビジネス行為の最後に行われます。既に契約が成立した後、またはすべての必要な処理が成り立つことが証明された後の契約直前なので、その恐れはありません。つまり、コンペーションは可能です。

現実には、別の考慮点も存在するでしょう。前述のように、厳密なACIDモデルではなく、コンペーションを前提としたモデルでは、そのビジネス取引上で売り手側は後にキャンセルを受けるかもしれない権利(商品)を一時的に手放すことにより、本来なら存在するビジネス・チャンスを失う可能性があります(仮在庫不足など)。しかし、これはないものがあると問題を起こすのに比較すれば、ビジネス上は安全側のプロセスです。ただし、このビジネス・ロスの最少化を目的とするなら、キャンセルが起こり得る確率を考慮して商品の量を仮増して考えれば解決できます。

要は、ビジネス契約上、万一オーバーブッキングが発生したときにどう対処するかの問題です。また、個数において絶対的制限がないもの、例えば生産可能な製品においては、万一オーバーブッキングが発生しても追加生産によってその場をしのぐことが可能です。この場合には、納入遅れに対する割り戻し金が発生するかもしれませんが、まさにそれがビジネス上の契約、つまりコンペーションなのです。

以上のように、一般的にはコンペーションを系統的に組み込むことは困難ではありません。また、それによって発生するビジネス・チャンスの損失やオーバーブッキングの問題も、B to Bがもたらす利便性を総合的に考慮すると十分に相殺し得るものと考えられます。

5.1.2. ビジネスACIDの実現可能性

コンペーションを前提にして、ビジネスACIDの可能性について検討します。ここでは、ビジネス・トランザクションを構成する個々のトランザクションにおいては、従来の意味でのACIDが保証されていることを前提とします。

まず、Atomicityについてです。ビジネス・トランザクションにおける意味は、それが終了した時点ですべてのビジネス的な目的が達成されているかを判断し、達成されなかった場合には相互の契約に基づいたレベルでビジネス的処置が行われていることであるといえます。当初の意図通りにビジネスが終了すれば問題ありません。ビジネス上またはシステム上のトラブルでそれが不可能な場合には、前述のように契約に基づいた履行義務が遂行される必要がありますが、これはまさに前述のコンペーション

そのものです。よって、ビジネスACIDは保証されます。

次に、Consistencyについてです。これは、トランザクションの終了時の状況にかかわらず、データの一貫性が保たれていることです。もともとこの問題は、参照の保全性や個々の売上高データベースと総売上高データベースのように、アプリケーション・プログラムの責任において考慮されるべき問題です。これは、ビジネス・トランザクションも同様です。ビジネス・トランザクションにおけるシステムの一貫性とは、ビジネス・トランザクションの終了時の状況にかかわらずビジネス的に一貫性が保たれることにほかなりません。ビジネス的一貫性を保つために、コンペーションが実行される可能性があります。そういう意味では、この問題はコンペーション時におけるアプリケーション・プログラムの考慮による一貫性の保証となります。

Isolationの本来の目的は、あるトランザクションの仕掛かり中の不確実な状況が、ほかのトランザクションに見られてシステム上での処理に矛盾が発生することを避けることです。ビジネス・トランザクションにおけるIsolationの目的は、B-UOW中にほかのビジネス・トランザクションからその仕掛かり中の状況を見ることができてビジネス上の混乱が発生しないことです。

B-UOWは、従来のUOWを複数含んだ構造なので、必然的に従来のUOWの観点から見ればUOW間はIsolationの保証はなくなり、B-UOW内においてはシステム的なIsolationは保たれていません。つまり、B-UOWの最終結果として在庫が100個になるかもしれませんが、途中の仮引き当ての結果中には90個に見えて、それに基づいてほかのビジネス・トランザクションが実行される可能性があります。

図7にそのイメージを示します。

この問題は、前章で述べたビジネス機会の損失とオーバーブッキングの問題と同様、対応としてコンペーションを伴う可能性があります。さらに、コンペーションを可能にするには、5.1.1節で述べたように契約成立前は購入品に手を付けないようにしておく必要があります。以上のような対応により、

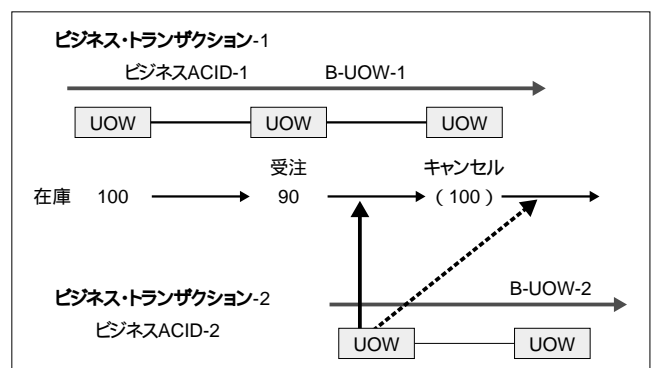


図7 . B-UOWとIsolation

表2 . ビジネスACIDとその対応

| | 従来トランザクション | ビジネス・トランザクション | 対応 |
|-------------------|--|---|-------------------------------------|
| Atomicity (原子性) | トランザクションの終了時にその処理のすべてが完了しているかまったく行われていないかのどちらか | ビジネス・トランザクション終了時にすべてのビジネス的な目的が達成されている。または未達時には契約に基づいたビジネス的処理が行われている | コンベンセーション |
| Consistency (一貫性) | トランザクション処理の終了状態にかかわらずデータベースの一貫性を保つ | ビジネス・トランザクションの終了時の状況にかかわらずビジネス的にデータベースの一貫性が保たれる | コンベンセーションとアプリケーション・プログラミング上の工夫 |
| Isolation (独立性) | 複数のトランザクションを実行したときにほかのトランザクションから影響を受けずに実行できる | 複数のビジネス・トランザクションを実行したときにほかのビジネス・トランザクションから影響を受けてビジネス上の混乱が発生しない | 機会損失、コンベンセーションとアプリケーション・プログラミング上の工夫 |
| Durability (永続性) | トランザクションの処理結果が永続的に保証される | ビジネス・トランザクションの処理結果が永続的に保証される | 従来ACIDに同様 |

Isolationに関するビジネス上の矛盾は避けられます。

最後に、Durabilityについてです。トランザクション終了後の状態を、システムなどの障害にもかかわらず永続的に保持できるかは、ビジネス・トランザクションだからといって特別な考慮は必要ありません。従来のトランザクション・レベルのACIDが保証できるメカニズムが存在すれば、同様に保証されます。

以上のビジネスACIDの定義とそれを実現するための対応に関して表2にまとめます。

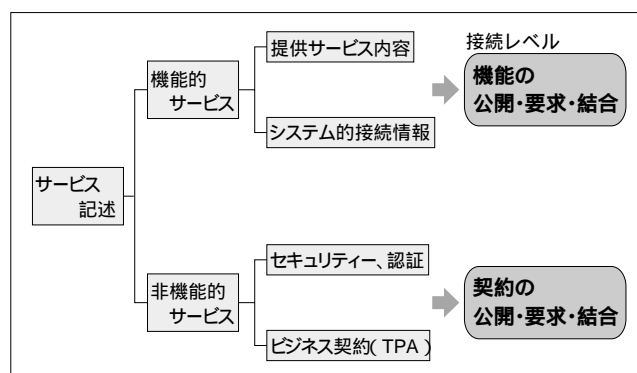


図8 . サービス記述と接続レベル

5.2. ビジネス・プロセスとしてのトランザクションの実現

ビジネス・トランザクションの基礎であるビジネスACIDの可能性は見てきましたが、ビジネス・プロセスとして実現可能かどうかを検討します。

5.2.1. ビジネス相手の選択

4章で述べたように、ビジネスの相手をダイナミックに見つけ、それにEnd-to-Endでつなぐレベルは達成しつつあるようです。しかし、現状のサービスの公開(記述)については、完成されているとは言い難い面があります。

そもそもサービス記述は、サービス要求者が自分の要求を満たしてくれるものかどうかを判断できるものでなければなりません[参考文献7]しかし、現時点でのサービス記述の仕様は、どのようなビジネス・サービスがどのようなシステム形態で接続できるかという、サービスの機能的な側面に主体が置かれています[参考文献8,9]。現実には、ある相手とビジネスを行うかどうかを判断し、実行するには、さらにサービスの非機能的な側面が必要です。「非機能的な面」とは、セキュリティ、プライバシー、および法的または契約上の問題でTPA(Trading Partner Agreement)といわれるものです。実際に、システム的につながった上でどのようにビジネスを行っていくかの取り決めと手続きであり、これなくしては実際のビジネスの遂行は不可能です[参考文献3]

この非機能のサービス情報の定義の位置付けは、現在のSOAP、UDDI、およびWSDLによって実現されているSOA(Service Oriented Architecture)のレベルが「機能の公開・要求・結合」であるのに対して、その完成度をより高めた「契約の公開・要求・結合」レベルであると筆者は考えます。

図8にサービス記述とその接続レベルをまとめます。

このようなサービス記述が実現されることにより、ビジネスの対象選択時に、サービスの機能面のみならずビジネス的に重要な非機能面も考慮され選択することが可能になります。

5.2.2. ビジネス・プロセスとしての実行

これまで説明したようなサービス定義が完成されることにより、実際のビジネス・プロセスに即したワークフローもシステムによって自動的に作成することが可能になります。これにはもちろん、コンベンセーションとその実行に必要なアプリケーション・ログも必要に応じて随時組み込まれます。このようにして完成されたワークフローにより、お互いに安全かつ確実にビジネスができるようになります。例えば、TPA上で取り決められた期限内に納入できなければ、そのビジネスは破棄され契約上の補償が行われます。システム的にいえばトランザクションのアポットと対応するコンベンセーションの実施です。

ここまで可能になってこそ初めて、ビジネス・トランザクション

の完成といえるでしょう。

5.2.3. ビジネス・データの交換

忘れてはならないことがもう一つあります。それは、結合されたアプリケーション間を流れるデータの表現形式の違いです。この問題は、4章で述べたようにXMLによって解決されます。Java™がプログラム・レベルでプラットフォーム独立を可能にしたと考えるなら、XMLはデータ・レベルでアプリケーションの独立を可能にしています。SOAP、UDDI、WSDLの基礎もXMLです。

ビジネス・トランザクションのACIDを保ちながらの実行は、コンベンションと従来のシステムによるACIDにより基本的に可能といえます。さらに、B to Bの大きなメリットでもあり目標でもある多様なサービスのダイナミックな取り込みについては、今ひとつ不完全な面があるのも事実ですが、基本的にはSOAP、UDDI、そしてWSDLによってその基礎は十分に提供されていて、今後その一層の充実が待たれる状態です。また、そのビジネス・トランザクション間での異種データ交換の問題もXMLにより解決されます。

よって、ビジネス・トランザクションは実現可能なレベルに達しつつあるといえます。

6. B to Bのシステム的な総括

B to Bについてシステム的に種々の側面から論じてきましたが、ここではそれを推し進めている源を少し俯瞰して考えてみます。なぜなら、それを知ることにより、今行われているB to Bへの対応が確かなものかどうか推し量れるからです。

コンピューターが計算機と呼ばれ、単なる高速の計算手段としてビジネスの世界において補助的に使われていた時代から、いつしかコンピューター・システムとして企業の業務の主力を担うようになりました。さらに、ネットワークと融合することにより企業活動全般をシステムによってコラボレーション可能な状態とし、企業の業務効率改善のみならず業務改革をも可能にできました。

今起きているB to Bは、その流れを一企業のみならず企業間にまで押し広げようとするものであり、経営的側面からのその要求は非常に強いものがあります。システム的なその原動力について、筆者は次のように考えます。

B to Bも、当たり前の話ですが一種のシステムです。システムはそれぞれが明確な機能を持ったコンポーネント群とそれらの相互作用によって成り立っています。システムがシステムとして成り立ち得るには個々のコンポーネントの優秀さも重要ですが、さらに大切なことは、そのコンポーネント群の相互作用を可

能にする機能が完備していることです。生物も実に完成されたシステムであり、胃・心臓・腎臓などの明確な機能を持ったコンポーネントがあり、インターフェースも明快であることによりコンポーネントの差し替え、つまり臓器移植を可能にしています。こうしたコンポーネントが真の実力を発揮できるのは、それらを結ぶ神経系統が極めて優秀なためです。いくら優秀なコンポーネントを持っていても、神経系統が侵されるとシステムとしてまったく機能しないことが、それを証明しています。

B to Bも同様で、各企業の多様なシステムが持つサービス(コンポーネント)を企業間でいかに連携し相互作用させるかが課題でした。その方法として考えられるのが、一つは多様性を排して同一システム構造で統一する方法です。もう一つの方法は、多様性を前提としてあるレベルで共通なシステム構造を導入することです。

前者の方法は、システム構造の低いレイヤーで行われるほど効率が良いでしょう。なぜなら、低いレイヤーほど求められる機能は共通かつシンプルで統一しやすく、その広がりも膨大であり、統一の効果が大きく出るからです。反対に、より高いレイヤーほど企業固有の要求が強く、無理に統一しようとする時間とコストの大きな問題に直面します。この場合は、統一は避け、あるレイヤーの上に統一用のレイヤーを設け、そこを連携の共通基盤とするのが賢明といえます。

前者のアプローチは、ネットワーク層のIPの採用によってインターネット・システムとして花開きました。後者のアプローチを探ろうとしているのが、まさにXMLを基礎とする各種技術の採用です。しかも、それらはプラットフォーム、オブジェクト・モデル、言語独立で共通基盤としての要求をよく満たしています。

このように、現在行われようとしているB to Bのシステム的アプローチは、実に合理的かつ現実を見据えたものです。さらに、B to Bの狙いであり存在意義でもあるダイナミックe-businessへの対応も、SOAによってアーキテクチャ的に裏打ちされて

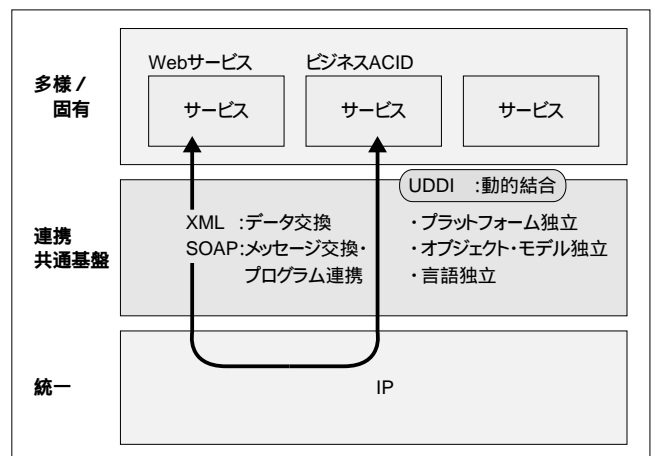


図9. B to Bフレームワーク

います(図9)。

このように、B to B の企業間をダイナミックにEnd-to-Endで結ぶ件に関しては、大きく一歩踏み出したといえます。

最終目的であるビジネス・トランザクションの実現、つまりビジネスACIDとビジネス・プロセスについては、コンベンションとの組み合わせにより、基本的に可能です。しかも、そのコンベンションのB to Bシステムへの組み込みに関しては、コンベンション自体がビジネス遂行の取り決めであり、その情報も今後サービス記述の一つとして実現されていくでしょう。トランザクションの側面としてのもう一つの懸念は、SOAPにはトランザクショナルなメッセージの交換についての仕様が現時点では含まれていないことです[参考文献12]

われわれSEにとっての現時点の仕事は、SOAPの持つ大きな利点とその特性をよく理解し、システム構築においてそれを利用することです。最終的には、ビジネス・プロセスとして企業間統合したワークフローを完成させることであり、しかもそのワークフローを作る仕組みが製品やサービス・レベルで提供されることですが、そのための基礎技術は既に整いつつあり、その発展も目覚ましいものがあります。もはや、少しの知恵と努力により、ダイナミックe-businessの先行者利益の享受の道を選ぶ時期に達しているといえます。

7. おわりに

ITを使用したシステムは、初期のコンピューターの出現以来数多く作られ、それぞれの時代において企業/社会変革に大きな役割を果たしてきました。しかし、それらのシステムは、広がりや有効性という側面から見るとまだ限定されたものでした。つまり、サービスの享受者が限定されていました。今や、それがWebサービスとしてだれもが容易に使えるユーティリティーになろうとしています。

これまでのユーティリティーの典型である電気やガス、通信設備なども、われわれ人類に計り知れない恩恵を提供してきましたが、それらはレイヤー的に見ると基盤的なものです。しかし、現在はB to Bの発展形態であるWebサービスによって統合された、より高いレベルのサービスがーから作られるのではなく、既存の資産を活用して一挙にユーティリティー化されようとしています。別の意味でいえば、われわれ人類は新しいユーティリティーとして素晴らしい資産を手に入れようとしています。

人類は、今後多くの困難な問題を克服しながら次世代を切り開いていかなければなりません。この新しい社会資産はその大きな助けとなるでしょう。また、そのためにもわれわれIT

プロフェッショナルは、その実現に向けて多くの貢献しなければなりません。本論文がその一助となれば幸いです。

[参考文献]

- [1] Asit Dan and Francis Parr "COYOTE HPTS Workshop 1997", IBM T. J. Watson Research Center
- [2] "SAGA 1987 ACM Hector Gracia-Molina", Kenneth Salem Princeton University
- [3] A. Dan, D. M. Diass 'Business to business integration with tpaML and BPF' "IBM Systems Journal", IBM Research
- [4] XML <http://www.w3.org/XML/>
- [5] Dan Gisolfi Solutions Architect 'An introduction to dynamic e-business', IBM jStart EmergingTechnologies (<http://www-106.ibm.com/developerworks/webservices/library/ws-arc/>)
- [6] JDDI Executive White Paper, September 2000
- [7] Burbeck The Path to e-business 2000/10. Emerging Technologies", IBM Software Group
- [8] "UDDI V2. Data Structure Reference UDDI Open Draft Specification", 8 June 2001
- [9] "Web Services Description Languages(WSDL) 1.1.W3C Notes", 15 March 2001
- [10] IBMプレスリリース(<http://www-6.ibm.com/jp/NewsDB.nsf/2001/05161>)
- [12] 野村 幸平「分散ウェブアプリケーション開発基盤としてのSOAPの有用性」2000年度IBMプロフェッショナル論文

(ページ数および表記上の観点から、著者の了解を得て編集部にて手を入れてあります)