

〈論 文〉

プラットフォーム・ソフトウェア市場への新規参入の成功要因

— 「スタックの破壊」と既存事業者と異なる「レイヤー優先度」 —

根 来 龍 之 *

佐々木 盛 朗 **

Factors Contributed to Successful Entries into a Platform Software Market

— Disruption of Stack and Different Layer Priority from Incumbents' —

Tatsuyuki Negoro

Shigero Sasaki

Abstract

What factors contributed to successful entries into the market of a software product that is used as platform? One of the characteristics of platform products is that the users do not enjoy their value until they are used together with complements or other platform products. And the entrants are likely to beat the incumbents by disruptive innovation (Christensen and Raynor, 2003). This paper describes the case of entries into the market of database with the related platform products. By low-end disruptive innovation, Oracle and Microsoft succeeded in their entries around 1980 and 1990, respectively. MySQL failed in its entry even though it brought a low-end disruption in the late 1990's. This case indicates that a *disruption of stack* and different *layer priority* from incumbents' may be factors contributed to successful entries into a platform software market.

要 約

プラットフォーム (PF) 製品として使用されるソフトウェアの市場への新規参入の成功要因は何であろうか。PF 製品の特徴の一つは、それ自体では利用者に価値を提供せず、補完製品および他の PF 製品と組み合わせあって価値を提供することである。一方で、新規参入者が既存事業者を打ち負かす可能性が高いのは、破壊的イノベーションが有効な状況であろう (Christensen and Raynor, 2003)。共に用いられる PF 製品の新規参入にも触れつつ、本稿ではデータベース市場への新規参入の事例を分析する。Oracle 社と Microsoft 社はそれぞれ 1980年と1990年の前後にローエンド型の破壊的イノベーションを起こして参入に成功したが、1990年代後半に同様にローエンド型破壊を起こした MySQL 社は参入に失敗した。この事例より、「スタックを破壊する」と「既存事業者と異なるレイヤー優先度を設定する」ことの二つが、PF ソフトウェア市場への新規参入の成功要因となることを示す。

* 早稲田大学大学院商学研究科 教授

** 日本電気株式会社 サービスプラットフォーム研究所

1. はじめに

プラットフォーム (PF) 製品として使用されるソフトウェアの市場への新規参入の成功要因は何であろうか。コンピュータ産業においては、1979年には IBM 社などの垂直統合型の企業の時価総額が大きかったが、2005年には Microsoft 社などの水平分業を担う企業の時価総額が大きくなった (Baldwin, 2007)。これは、オペレーティング・システム (OS) やデータベース (DB) などの、PF 製品であるソフトウェア (PF ソフトウェア) における成功が企業価値の大きな増加につながるということでもある。本稿の目的は、DB 製品の事例分析から、PF ソフトウェア一般に通じるとされる新規参入の成功要因を探ることである。なお、本稿での参入成功は、ある年度において事業を黒字化し、かつ市場の寡占企業の一社になることである (寡占の定義は後述)。

PF ソフトウェアの特徴の一つは、それ自体だけでは利用者に価値を提供せず、補完製品および他の PF 製品と組み合わせることで価値を提供することである。この特徴は、PF ソフトウェアだけではなく基盤機能型 PF (根来・加藤, 2010) 一般に共通の性質であるが、例えば、OS の製品である Windows を使って文書を作成するには、補完製品となるアプリケーション (AP) の Wordなどを動作させる必要がある。また、OS を動作させるためには、PF 製品である PC などのハードウェア (HW) 製品が必要である。つまり、AP と、OS、HW のレイヤーを積み重ねて、Word (AP)+Windows (OS)+PC (HW) のスタックを利用者が構成し、使用するのである。ここで、Windows (OS) と PC (HW) は PF 製品ということになる。サーバー・アプリケーションを実行するのであれば、AP と OS の間に DB のレイヤーを入れることも一般的である。この場合は、DB、OS、HW が PF 製品となる。

新規参入者が既存事業者を打ち負かす可能性が高いのは、破壊的イノベーションが有効な状況であろう (Christensen and Raynor, 2003)。破壊的イノベーションは、「非消費者に対して新しい機能特徴をぶつけることによって、新たなマーケットを創造するか、あるいは既存のマーケットのローエンドにいる顧客に対してそれまで以上の利便性が低い価格を提供する」(Christensen *et al.*, 2004) ことで、最初は小さな市場を得て、徐々に性能向上とともに既存技術を代替して自分の市場を大きくしていく力を持つたぐいの技術革新を意味する。そのため、新規顧客や既存業者にとって魅力のない顧客群に安く売れる、シンプルで便利な製品を低価格で商品化することが課題である状況では、破壊的イノベーションによる新規参入が成功しやすい。しかし、この種の新規参入が常に成功するのでなければ、そのような状況下で参入に失敗する原因は何か (Danneels, 2004)、という問いが生じる。Christensen らが、破壊的イノベーションの事例として詳細に扱ってきたのは、ハードディスク (Christensen, 1997) や鉄鋼製品 (Christensen and Raynor, 2003) などの非 PF 製品 (それ自体で価値を持つ製品) であった。本稿の問題意識には、PF 製品に固有の特徴を考慮することで、破壊的イノベーションに関する議論に新たな知見を加えることが含まれる。

本稿では PF 製品の一種である DB の市場への新規参入の事例分析を行う。この際に、DB と共に用いられるその下の階層 (レイヤー) の PF 製品の市場への新規参入も同時に考慮する。本稿で対象とする既存事業者と新規参入者の組み合わせは、IBM 社と Oracle 社、Oracle 社と Microsoft 社、Oracle 社と MySQL 社、Microsoft 社と MySQL 社の四組である。また、DB 市場に加えて、OS 市場と HW 市

場への新規参入にも触れる。そして、市場の概況、価格、性能に関する情報を提示し、複数のレイヤーにわたる競合関係を整理する。DB 市場を中心に扱う理由は、新規参入に複数の企業が成功して、寡占市場が形成されたことと、PF ソフトウェアである DB 製品が、同じく PF ソフトウェアである OS 製品の上で動作する補完品であるという PF 製品自体の階層性が観察できるからである。なお、本稿では、AP、DB、OS、HW などの製品グループの各グループを、レイヤー（階層）と呼ぶ。階層性とは、上のレイヤーの製品が動作するために、下の製品の存在を前提にするが、逆は成り立たないことを言う¹⁾。

Oracle 社と Microsoft 社はそれぞれ1980年と1990年の前後にローエンド型のイノベーションを起こして参入に成功したが、1990年代後半に同様にローエンド型イノベーションを起こした MySQL 社は参入に失敗した。ローエンド型のイノベーションが破壊的イノベーションでありえる条件の一つは、「既存企業に収益をもたらしてはいるが、過剰満足である顧客に対して従来の基準に沿った満足できる性能を低価格で提供するイノベーション」であることである (Christensen *et al.*, 2004)。ここで、過剰満足とは、該当顧客が必要とする以上の性能を既存製品が持つことを意味する。MySQL 社の製品もこの条件を満たしていたが、同社の新規参入は成功しなかった。このことは、Christensen らが触れていない、「ローエンド型の破壊的イノベーション」以外の成功要因があることを示唆している。

本稿が結論として示す PF ソフトウェア市場への新規参入の成功要因は、「スタックを破壊する」とことと「既存事業者と異なるレイヤー優先度を設定する」とことの二つである。スタックの破壊とは、スタック全体としてローエンド型の破壊的イノベーションを起こすことである。ただし、スタックの全ての構成要素を破壊しなければ、スタックを破壊できないわけではない。例えば、高価な DB と、OS、HW 及び安価な OS と HW が提供されているとする。このとき、安価な DB を提供できれば、既存の AP+DB+OS+HW のスタック全体に対して安価なスタックが成立する。一方、レイヤー優先度とは、例えば DB と OS の両方の製品を提供する Microsoft 社が、どちらのレイヤーの売上を優先させるかを表す。

2. 先行研究

本稿で対象とする新規参入戦略の出発点は、破壊的技術 (Bower and Christensen, 1995、Christensen, 1997)²⁾ による参入である。破壊的技術は、主流市場での性能指標では劣っているが、異なる指標においては優れる。よって初期の製品はニッチ市場でしか受け入れられないが、該当技術の改善によって主流の性能指標が向上するにつれて主流市場の顧客を奪っていく。

Adner (2002) は、「事前に」技術が破壊的かどうかを判定するために、技術が破壊的になりやすい状況を研究している。ある顧客セグメントで重視される性能指標が他のセグメントでも重視される程度である嗜好重複 (preference overlap) と、その対称性を表す嗜好対称性 (preference symmetry) の二つの概念を導入する。嗜好重複が小さい場合には、一方のセグメントで購入されている製品が他方でも購入される可能性は小さい。十分に嗜好が重複し、かつ対称的であれば、製品間の競争条件は収束していく。非対称の場合は、破壊的な技術が生まれやすいとされる。

Adner (2002) などの破壊的技術に関する研究に続いて、Danneels (2004) は、破壊的技術に関して議論すべきテーマを提示している。これらは、破壊的技術の定義、企業または産業の将来の予測可能性、

破壊的技術が出現しても成功している既存事業者、破壊的な技術の変化の下でも顧客指向であることのメリット、破壊的技術のためにスピノフを作ることのメリット、の五つに分類されている。Danneels (2004) に関連して、Govindarajan and Kopalle (2006) は、イノベーションの不連続性 (radicalness) は技術の新規性の程度を示し、破壊性 (disruptiveness) はイノベーションが導入された時点での新たな顧客セグメントが認識する価値の程度を表すものだと指摘しつつ、カニバライゼーションを恐れない企業は破壊的イノベーションを生みやすいと主張している。

上記の「破壊的な技術 (イノベーション)」に関する議論は、対象とする製品として PF 製品に特に焦点を当てているのではないため、本稿の先行研究として PF 製品に関する議論にも触れておく必要がある。基盤機能型 PF 製品 (根来・加藤, 2010) には、補完製品の開発者と利用者という二つのサイド (プレイヤーグループ) が関与する。このように、複数のサイドが関与する市場はツーサイド市場と呼ばれる。ツーサイド市場で重要な問題として、Eisenmann *et al.* (2006) は、収益格差のマネージメント (助成サイドと収益サイドの存在) と、一人勝ちの力学、プラットフォーム包囲 (platform envelopment) の脅威、の三つを挙げている。

異なるサイドの収益格差のマネージメントに関する研究である Paker and Van Alstyne (2005) は、消費者と開発者のサイドについて、それぞれの価格と販売数量の関係を考慮して、どちらを助成すべきかを考察している。この決定は、あるサイドの数が増えると別のサイドの数も増えるというサイド間ネットワーク効果を考慮して行われるべきだとされる。一人勝ちの力学に関連する重要な概念であるマルチホーミング・コストは、「複数の PF を利用する際の追加コスト」として、Eisenmann (2007) で定義されている。Eisenmann *et al.* (2007) は、ある市場で支配的な PF を持つ企業が、その PF と補完品をバンドリングさせて、補完品の市場へ参入する戦略としての、「プラットフォーム包囲」について述べている。

PF 製品の特徴を反映した新規参入戦略はプラットフォーム包囲だけではない。加藤 (2009) では、Java の事例に基づいて、PF 間競争における施策である「階層介入」の概念が提示されている。階層介入は、市場での形勢が劣勢である企業がクロス PF 製品 (階層) の投入 (介入) を行うことを指す。根来・加藤 (2010) は、プラットフォーム包囲における上位レイヤーの包囲と下位レイヤーの包囲を区別している。そして、階層介入を platform bridging に一般化しつつ、platform compatibility と platform alliance の戦略を一人勝ちに対抗する戦略として追加している。また、PF は補完品を使用するための基盤機能と仲介や決済などのメディア機能を併せ持つことも指摘されている。

本稿の特色は、破壊的イノベーションと PF 製品の特徴の双方を考慮した新規参入の成功条件を論じる所にある。さらに、PF ソフトウェアを対象にすることから、ソフトウェアビジネスに関する研究と、PF 製品はモジュール製品の一種であるという意味でモジュール化に関する研究と、本稿は関連する。

ソフトウェアビジネスに関する先行研究として、國領 (1999)、Varian (2000)、West (2003) を挙げる。國領 (1999) は、変動費がゼロないし極めて小さく、固定費だけが存在するなら、固定費を回収した後には財やサービスの無償提供が成立しえることを指摘している。Varian (2000) は、情報財の性質には、経験財であること、高い固定費と低い複製費を持つこと、公共財であること、の三つがあると指摘

している。West (2003)は、オープンソース・ライセンスは、ソフトウェアに対する研究開発投資の直接回収をあきらめる代わりに、多く利用してもらうことで、直接ソフトウェアから利益を上げられなくとも、関連する HW や、サポート、トレーニングなどの補完品の販売につなげようとしていることを指摘している。

モジュール製品に関する先行研究としては、以下が本稿の議論に関係する。Baldwin and Clark (1997)は、IBM 社の System/360の開発において、モジュールの協調動作を規定する「可視的な」情報と、モジュール内の動作に関する「隠された」情報が認識されていたことに注目している。また、Schilling (2000)は、特定のコンポーネントの結合によって優れた機能を実現できる (synergistic specificity) 場合には統合化が促進されること、市場に参加する企業が異なる能力を持つほど異なる企業によるモジュール化が促進されることなどを述べている。Baldwin and Woodard (2008)は、プラットフォーム・システムの構成要素は、多様で変化しやすい補完品と、変化しにくいコア、コアと補完品を一つのシステムとして機能させるためのデザインルールの三つであると述べている。

3. データベース事業の事例

本章では、データベース事業の事例を分析する。分析の時点は2009年7月末である。3.1節では、データベース管理システムをプラットフォームとしての観点から説明する。3.2節では、寡占状態であるデータベース市場の概況について述べる。3.3節では、データベースと関連する製品の価格の情報を提示する。3.4節では、DB、OS、HW の三つのレイヤーにわたる競合関係を分析する。

3.1 プラットフォームとしてのデータベース管理システム

データベース管理システム (DBMS) は、構造化されたデータの集合を管理するソフトウェアである。DBMS はデータの追加や参照、加工を行うための機能を提供する。AP の開発者にとって、データの検索、更新、挿入、削除を正確に行うのは、同一データへの同時アクセスや、処理の中断、十分な性能の維持などを考慮すると、複雑な作業である。この処理を容易かつ効率的に行うために、DBMS が必要とされるようになった。DBMS にはいくつかの種類があり、これらを総称してデータベース (DB) と呼ぶ。汎用性の高さなどから、現在主流の DB は関係データベース管理システム (RDBMS) である。

Oracle や、SQL Server、MySQL などの DB を動作させるためには、Sun 社の SPARCstation などのミッドレンジ・サーバーや、Intel 社の CPU を搭載した IA サーバー、PC などの HW に加えて、ほとんどの場合は Unix の Solaris や、Windows、Linux などの OS が必要である。つまり、ERP や、BI、動的サイトの生成などの AP は DB の機能を利用するが、DB はプロセス管理などの OS の機能を利用する。そして、OS は HW 上で動作する。本稿では、この関係を AP+DB+OS+HW のように記述する。そして、このような依存関係で結びつけられた AP を実行するための製品群は、一般にスタック (stack) と呼ばれる (図 1)。

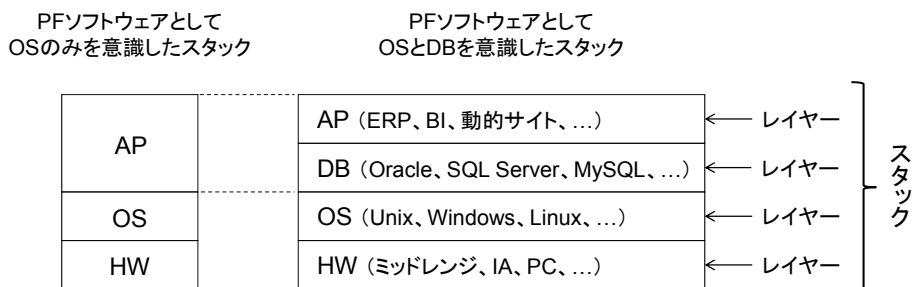


図1 レイヤーによって構成されるスタック

多くの人々にとって最も馴染み深いスタックは、Word+Windows+PC といった、デスクトップ・アプリケーションの AP+OS+HW のスタックであろう。AP は利用者が直接望む機能を提供するプログラムであり、OS は AP に対して統一された Application Program Interface (API) を提供する。OS があることを前提として API を利用する AP を開発することには二つの利点が含まれる。第一に、HW の機能を直接使用するよりも、OS が提供する変化しにくい機能 (Baldwin and Woodard, 2008) を使用することで容易に高度な機能を実現できる。第二に、ある特定の OS 向けに AP を開発すれば、その AP は OS が対応している HW 上であれば動作する。このように、OS を前提として AP を開発することと、その OS が前提とする HW を含めて、AP を実行するためのスタックを構築することは、十分効率的なのである。

OS から見れば自身の機能を利用して動作するソフトウェアは全て AP であるが、AP 間での機能の依存関係も存在する。デスクトップのスタックで DB を実行することは少ないが、サーバー・アプリケーションのスタックでは DB が実行されることが多い。このとき、DB は OS にとっては AP であるが、DB の機能を利用する AP にとっては PF ソフトウェアである。そして、DB を AP として扱う AP+OS+HW というスタックではなく、AP+DB+OS+HW というスタックが構成される。また、ウェブ・アプリケーションのスタックでも DB を使うことを注記しておく。

3.2 市場の概況

本稿では、前述したように、参入成功の基準を、ある年度において事業を黒字化し、かつ市場の寡占企業の一社になることとする。黒字化できない事業参入は失敗と解する。また、ネットワーク効果の高い製品であるソフトウェアでは、寡占企業の一社にならなければ、長期的な視点からは成功とは言えないと考えることにする。PF にはしばしば一人勝ちの力学 (Eisenmann, 2007) が働くため、大きな売上高のシェアを確保しないと、長期的な事業の存続は難しいだろう。黒字化と寡占企業化を同時に達成しないとならないのは、採算割れしてもシェアを追う参入は、成功とは認めないためである。

寡占の目安は、売上高のシェアの75%以上を上位三社以内が占めることとする。この合計シェアの75%への参加企業を、本稿では寡占企業と呼ぶ。公正取引委員会 (2009) によると、企業結合後のハーフィンダール・ハーシュマン指数 (HHI) が2500以下であり、かつ企業結合後の当事会社グループ

の市場シェアが35%以下の場合には、競争を実質的に制限することとなるおそれは小さいと考えられている。HHI は各事業者の市場シェア (%) の二乗和である。また、Henderson (1976) の経験則³⁾によると、典型的には上位三社のシェアの比は4 : 2 : 1になり、その他の事業者は競争が困難になる。過渡的な状態として、*i* 番目にシェアが大きい事業者のシェアを s_i 、(*i*+1) 番目にシェアが大きい事業者のシェアは ($s_i/2$) と想定すると、HHI が2500以上になるのは、上位三社のシェアの和が75%以上の時である⁴⁾。ただし、三位の事業者のシェアは10%以上とする。

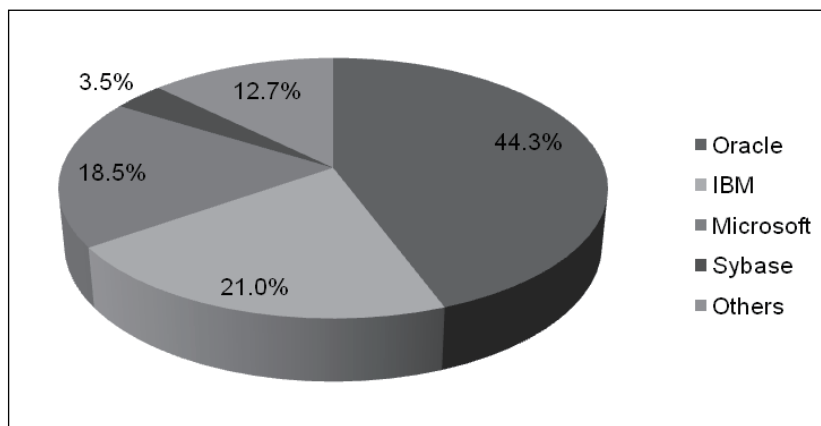


図2 売上高による製品シェア (Olofson, 2007)

Oracle 社、IBM 社、Microsoft 社の三社は、2007年の RDBMS 市場を寡占していた。IDC の調査 (Olofson (2007)) によると、2007年度の RDBMS の市場は、188億ドルの規模を持ち、前述の三社が 83.8%のシェアを占める寡占市場であった。内訳は、Oracle 社が44.3%、IBM 社が21.0%、Microsoft 社が18.5%であった (図2)。第四位の Sybase 社のシェアは3.5%であった。

Oracle 社のデータベース事業は、2007年において黒字であったとみなせる⁵⁾。2007年においては、Oracle 社の営業利益率は33%で、財務報告では、ソフトウェア事業とサービス事業を区分して管理している。ソフトウェア事業の粗利率は約60%であり、「データベースとミドルウェア」と「アプリケーション」の二つの製品セグメントがある。売上高の過半を占める「データベースとミドルウェア」に含まれるソフトウェアは、データベース、ウェブアプリケーションサーバー、ビジネスインテリジェンスなどである。2007年度の RDBMS の市場は188億ドルの規模を持ち、その44.3%を占める Oracle 社は約83億ドルを売り上げているため、セグメントの総売上高の約94億ドルのほとんどはデータベース事業の売上高であることがわかる。よって、データベース事業単独も黒字であったと考えられる。売上高の半分以上の比率を占める事業が赤字で60%前後の粗利率を達成できるとは考えにくいからである。

Microsoft 社のデータベース事業も、2007年において黒字であったと考えられる⁶⁾。Microsoft 社は、「クライアント」、「サーバーとツール」、「オンラインサービス」などの五つのセグメントを財務報告で区別している。「サーバーとツール」セグメントには、Windows Server、SQL Server、Visual Studio

などが含まれ、その売上高は約110億ドル、営業利益率は約33%であった。DB 市場の18.5%を占める Microsoft 社のデータベースである SQL Server の売上高は、約35億ドルである。仮にデータベース事業がブレイクイブンをであったとすると、75億ドルの売り上げで40億ドル弱の営業利益をあげたことになってしまうので、SQL Server 単独で黒字になっていると解釈できる。本稿で扱う最も古い既存事業者の IBM 社は別にして、以上から、この時点での参入成功企業は、Oracle 社と Microsoft 社であると解釈する。

一方、MySQL 社の取支が黒字化したのは、2006年のことだったと伝えられている⁷⁾。2006年の売上高は約0.5億ドルであった。黒字化が報じられたということは、それまでは赤字であったからだと考えられる。なお、MySQL 社は2006年以降において、市場の寡占企業の一社になりえたことはない。

3.3 価格

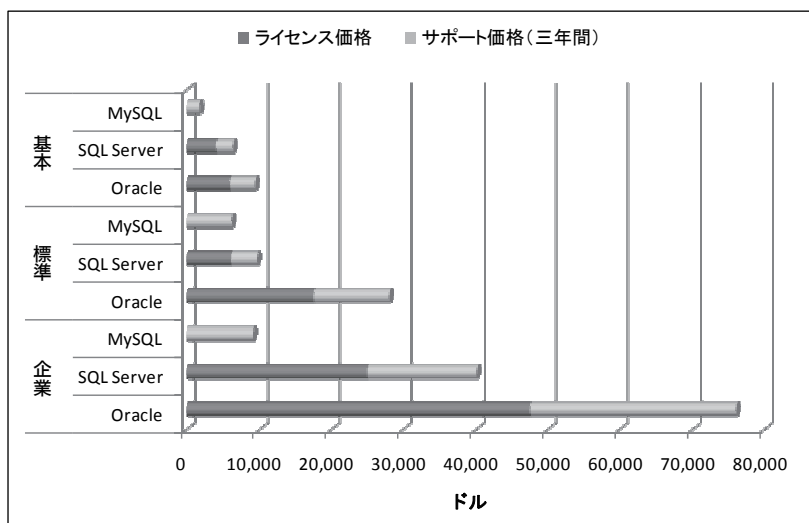


図3 エディション別のRDBMSの価格比較

1つの演算装置上での Oracle⁸⁾、SQL Server⁹⁾、MySQL¹⁰⁾の基本、標準、企業エディションのライセンス価格と三年間のサポート価格を、図3に示す。エディションの区分は Komo (2005)に従って、中小企業向けの「基本」、中規模の企業向けの「標準」、大企業向けの「企業」とした。エディションが異なれば、価格と、利用できる機能、利用できるハードウェアに課される制限などが異なる。基本的には、総額は演算装置の数に比例するが、何を一つの演算装置とするかは製品による。

RDBMSの導入にかかる費用には、エディションと演算装置数以外にも多くの要因が影響する。クラスタリングやパーティショニングなどの高度な機能を利用するには、オプションのライセンスを購入する必要がある場合がある。また、前述のサポート価格を支払うことで得られるのは、プログラムのアップデート、パッチの提供、24時間対応の問い合わせ対応などであり¹¹⁾、データベースを運用または管理するための全ての費用がカバーされるわけではない。

DBを利用する際に必要とされる、OSの価格にも言及する。Microsoft社のサーバーOSであるWindows Server 2008のライセンス価格は、1,029ドルから3,999ドルにわたる¹²⁾。IBM社のUnixであるAIXのライセンス価格は、CPU (Power V6.1)あたり2,600ドル、CPUあたりの3年間のサポート価格は2,216ドルであることが、DBの代表的なベンチマークであるTPC-Cの結果において公開されている¹³⁾。RedHat社は無料のOSのLinuxのサポートを提供しており、その価格は1,000ドル程度である¹⁴⁾。

表1 コンピュータの性能と価格 (Patterson and Hennessy, 2004)

年	機種	性能(加算/秒)	価格	(物価調整後)	価格性能比
1951	UNIVAC I	2,000	\$1,000,000	\$6,107,600	1
1964	IBM S360/model 50	500,000	\$1,000,000	\$4,792,300	318
1965	PDP-8	330,000	\$16,000	\$75,390	13,135
1976	Cray-1	166,000,000	\$4,000,000	\$10,756,800	51,604
1981	IBM PC	240,000	\$3,000	\$5,461	154,673
1991	HP 9000/model 750	50,000,000	\$7,400	\$9,401	15,122,356
1996	Pentium Pro PC	400,000,000	\$4,400	\$4,945	239,078,908
2003	Pentium 4 PC	6,000,000,000	\$1,600	\$1,600	11,452,000,000

RDBMSなどのソフトウェアを利用するために必要な、ハードウェア(コンピュータ)の価格と性能(Patterson and Hennessy, 2004)を表1に示す。基本的に、コンピュータの性能向上は、その上で動作するDBの性能向上を意味する。さらにHWの価格情報を追加する。IBM社のメインフレームのSystem/370 Model 135の価格は、1971年3月に発表されたものは¹⁵⁾、メモリのサイズによって475,000ドルから1,068,000ドルであった。通信線は9,880ドルから29,650ドル、ディスクストレージは23,465ドルであった。DECのミニコンであるVAX-11/780は1978年に出荷され、価格は120,000ドルから160,000ドルであったと伝えられている¹⁶⁾。これは、OSのVAX/VMS込みの価格である。一方、Sun社のコンピュータは、80年代後半に実用的な性能を40,000ドル程度で実現できたと伝えられている¹⁷⁾。機械統計年報(経済産業省, 2008)によれば、IAサーバーは5,000ドル程度、PCは1,000-2,000ドル程度であった。

3.4 新規参入時の競合関係

IBM社とOracle社、Oracle社とMicrosoft社、Oracle社とMySQL社、Microsoft社とMySQL社の四組の既存事業者と新規事業者の組み合わせについて、DB、OS、HWの三つのPFにわたる競合関係を分析する。主たる目的は、AP+DB+OS+HWのスタックにおける、DBの新規参入を整理することである。よって、特に着目するのはDBであるが、OSとHWの新規参入についても触れる。

Oracle社は1979年にデータベース市場に参入した(図4)。図4において、濃い影は新規参入者の製品を表し、薄い影は既存事業者の製品を表す。破線はその上下の製品を同一事業者が提供していることを表し、矢印は新規参入を表す。3.3節で示した情報に基づいて、おおよその製品価格も示す。個別の

要件が価格に大きく影響するので、ここでの価格はあくまで目安であるが、数百万ドルの製品か、数十万ドルの製品か、という価格帯の比較に用いるために提示する。

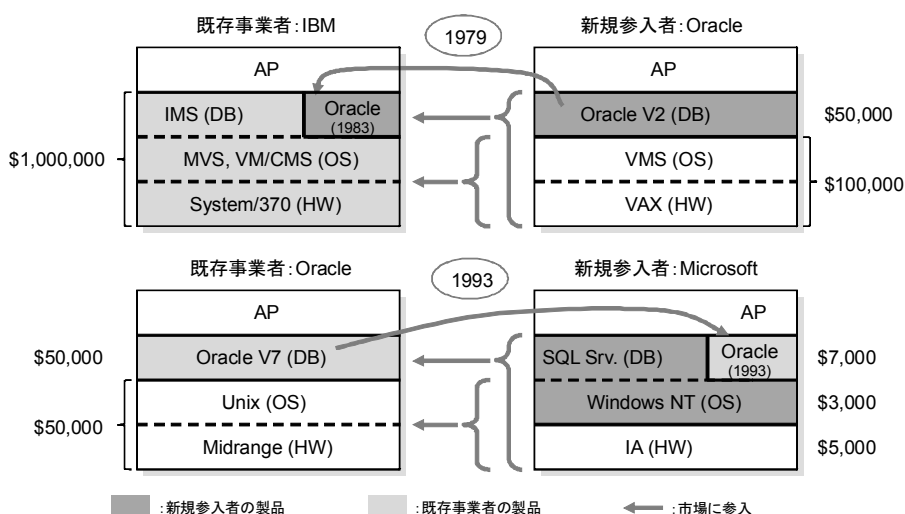


図4 Oracle社とMicrosoft社による新規参入

1979年当時、DBを利用する場合には三つのIBM社の製品からなる1,000,000ドル程度のスタックを利用することが多かった。これらの製品は、メインフレームのSystem/360の後継機種であるSystem/370、そのOSであるMultiple Virtual Storage (MVS)またはVM/CMS、階層型データベースのIMSである。このAP+DB (IMS)+OS (MVSまたはVM/CMS)+HW (System/370)という既存スタックに対して、1978年に発売されたDECのミニコンは、100,000ドル程度のVMS+VAXでスタックのOS+HWの部分に参入した。VAXはSystem/370よりも演算性能は低かったし、VMSもMVSまたはVM/CMSよりも性能が低かったと言っていいだろう。OSの性能は測るのが難しいが、より多く、より長く使われているOSは、より良いOSとして認知されることが多い。

Oracle社は、VMS+VAX上で動作する50,000ドル程度のRDBMSを1979年に提供した。この価格はVMS/VAXと大きくは変わらない。IMSは階層型データベースであり、関係データベースのOracleと単純に性能を比較するのは難しいが、Oracleが基本的な機能であるトランザクション処理をサポートしたのは1983年だったため、IMSの方が高性能と判断してよいだろう。すなわち、1,000,000ドル程度で提供されていたIBM社のAP+DB+OS+HWスタックに対して、1979年に100,000ドル程度のAP+Oracle+VMS+VAXが利用できるようになったのである。その後IBM社は、1980年にRDBMSをOSに統合したSystem/38を投入した。一方のOracleは、System/370上のVM/CMS上でも動作した。

Microsoft社は1993年にRDBMS市場に本格的に参入した(図4)。SQL Serverの本格参入はWindows NTが登場した後だとみなす。Windows NTが発売された1993年に主流だったスタックの一つは、AP+Oracle+Unix+ミッドレンジ・サーバーだった。Oracleは50,000ドル程度(企業エディション)、Unixとミッドレンジ・サーバーも50,000ドル程度で提供されていた。Sun社のように、ミッ

ドレンジ・サーバーを提供している企業が自社サーバー上で動作する OS も提供していることが多かった。Windows NT は3,000ドル程度で提供されたため、Unix+ミッドレンジ・サーバーが50,000ドル程度であったのに対して、低性能ではあったが NT+IA サーバーは10,000ドル程度だった。

SQL Server (標準エディション) は7,000ドル程度で提供され、既存の100,000ドル程度の AP+DB+OS+HW のスタックに対して、数万ドル程度のスタックを構成する製品として提供された (SQL Server は情報系 DB の市場で広まっていったため、企業エディションではなく標準エディションの価格を取り上げる)。少なくとも2003年の時点¹⁸⁾では、Oracle のようなプロプライエタリ (=特定企業特有) なソフトウェアを提供する会社は、概して機能競争を好んでいた。Oracle 社の年表¹⁹⁾とウェブ上で公開されている同社の技術資料²⁰⁾と、Delaney (2000)と Microsoft 社が公開している技術資料²¹⁾から、SQL Server の機能は Oracle に遅れて追加されていたことがわかる。例えば1996年に上市された SQL Server V6.5では、行レベルロックの機能が追加されたが、Oracle が行レベルロックをサポートしたのは1988年だった。2000年の SQL Server 2000では XML に対応し、Online Analytical Processing (OLAP) の機能が拡充されたが、Oracle が XML 対応したのは1998年、OLAP を扱うデータベースハウスの機能は1997年からサポートされた。

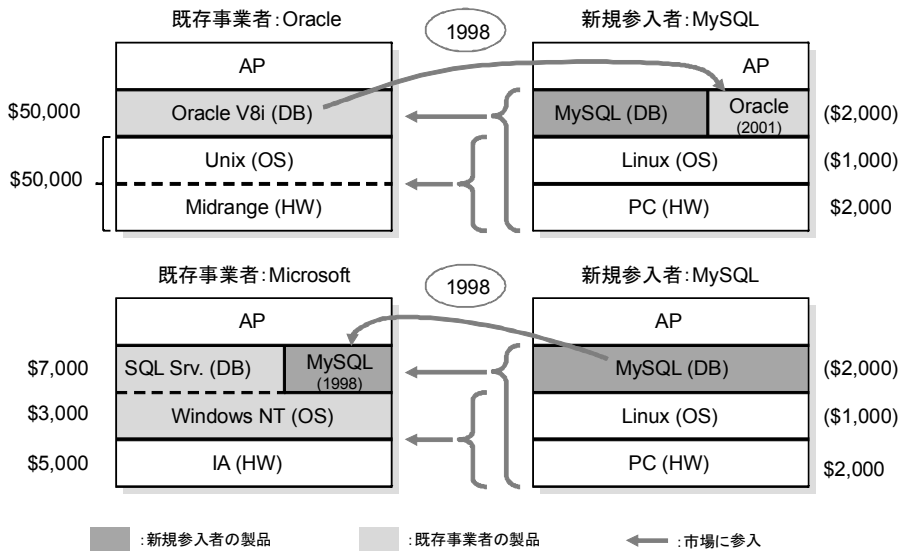


図5 MySQL 社の新規参入

サーバーOS 市場での NT のシェアはまだ小さかったが、NT 発売後の SQL Server は、NT 上でのみ動作した。一方、Oracle 社も NT+IA サーバーと近い価格帯の標準エディションを用意して、NT 上で動作する Oracle を1993年に投入し、情報系 DB の市場を新たに獲得した。これによって、AP+Oracle+Unix+ミッドレンジ・サーバーは依然として高価格であったが、AP+Oracle (標準エディション、20,000ドル程度)+NT+IA サーバーという安価なスタックが新たに成立した。

MySQL 社は1998年に RDBMS 市場に本格的に参入した (図5)。本格参入を1998年としたのは、

Windows 対応が済んだ年だからである。一方で当時は、PC の価格が2,000ドル程度にまで下がるとともに、無料の OS である Linux が台頭していた。Linux が最初に公開されたのは1991年9月であり、バージョン番号は0.01だった²²⁾。Linux 1.0は1994年3月に、複数のプロセッサを同時に利用できる Linux 2.0は1996年6月に、Intel の64ビットプロセッサをサポートする Linux 2.4は2001年1月に公開された。Linux 自体は無料だが、RedHat 社は1,000ドル程度の価格でサポートを提供した。Oracle は2001年に Linux 2.4に正式に対応したが、SQL Server は引き続き NT 上でのみ動作した。

2005年10月、MySQL のバージョン5.0の製品版がリリースされた²³⁾。このバージョンではカーソル、ストアドプロシージャ、トリガー、ビュー等の進んだ機能がサポートされた。Oracle がストアドプロシージャとトリガーをサポートしたのは1992年であった。ライセンス価格が無料である MySQL のサポートは、1,000ドル程度で提供された。この時点で既に、MySQL は、LAMP (Linux, Apache, MySQL, PHP) スタックの一部として組み込みまたはウェブシステム向けの DB として大きなインストールベースを獲得していた (Burgelman *et al.*, 2004)。サポートが必要なければ Linux+PC は2,000ドル程度であり、10,000ドル程度の Windows NT+IA サーバー よりも大幅に安価だった。そして、MySQL と Linux のサポート料を支払っても、AP+MySQL+Linux+PC のスタックは5,000ドル程度で構築でき、数万ドルの AP+SQL Server+Windows NT+IA サーバー のスタックと比べても大幅に低価格であった。

MySQL 5.0は、情報系向けの DB としても用いられるようになった。これは主として SQL Server のセグメントに進出したことを意味するが、MySQL 社への強い対抗策を講じたのは、主に基幹系向けの DB を提供していた Oracle 社だった。MySQL 5.0が公開された同月、Oracle 社は無料エディションを投入した²⁴⁾。Microsoft 社も無料エディションを同社に続いて投入した。MySQL は、DB のプリミティブな入出力のために、ストレージエンジンとしてモジュール化された Innobase 社の InnoDB を、トランザクション処理に関してはほぼ標準的に使用していた²⁵⁾。Innobase 社の InnoDB は、銀行口座間の資金移動のようなトランザクションを確実に処理するための機能を持つ。トランザクション処理向けのエンジンは、2005年当時はほとんど InnoDB だけと言える状態だった (Burgelman and Wittig, 2006)。MySQL 5.0の公開と同じ2005年10月、Oracle 社はこの Innobase 社の買収を公告した²⁶⁾。Sleepycat 社の Berkeley DB は MySQL のストレージエンジンに使うことができたが、2006年2月、Oracle 社は Sleepycat 社の買収も公告した²⁷⁾。同月に、Oracle 社から MySQL 社の買収の申し入れがあったと、MySQL 社の CEO がインタビューで応えたことが報じられた²⁸⁾。

2008年1月16日、Sun Microsystems 社が10億ドルでの MySQL 社の買収を公告した²⁹⁾。そして2009年4月20日に、Oracle 社が Sun Microsystems 社の買収を公告することになる³⁰⁾。

4. 本稿が設定する概念

本章では、データベース市場における破壊的イノベーションと競合関係を説明するために、それぞれ「スタックの破壊」と「レイヤー優先度」の二つの概念を設定する。4.1節では、スタックの破壊がスタック全体としてのローエンド型の破壊的イノベーションを意味することを述べる。4.2節では、どのレイヤーの製品の売上をより優先するか、を表すレイヤー優先度について述べる。

4.1 スタックの破壊

スタックの破壊とは、スタック全体としてローエンド型の破壊的イノベーションを起こすことである。利用者が利用したいのはAPであり、DBやOSなどのレイヤーのPFソフトウェアとHWは、あくまでAPを実行するために必要とされる。APを実行するためにはスタック全体が必要となるため、スタック全体を購入する利用者にとって問題となるのは、スタックとしての価格である。スタックを構成する全てのレイヤーで、既存品よりも安価な製品を提供しなければ、スタックを安価にできないわけではない。既存スタックの一部のレイヤーのみを残して、破壊されたレイヤーを使った新たなスタックが成立したときに、未破壊のレイヤーを破壊するだけでも、スタック全体を破壊することができる。レイヤー毎にその破壊は大きなリソースや非連続的なアイデアを必要とするため、スタックの全てのレイヤーを一度に破壊するのは非常に困難である。

未破壊のレイヤーを安価にすることでスタック全体を安価にする状況を、**図6**に例示する。既存の AP+PS2+PS1+PH のスタックが数十万ドルである一方で、数万ドルの AP+PS1+PH のスタックが提供されていたとする。この数万ドルのスタックが満足できる性能を提供している場合、AP+PS1+PH は既に破壊されたスタックである。しかし、AP+PS2+PS1+PH のスタックは破壊されていない。既存スタックのPS2の製品は、破壊されたスタック上で利用するには高価すぎるのである。安価で満足できる性能を持つPS2の製品を提供することで、既存の AP+PS2+PS1+PH のスタックを破壊できる。

未破壊のレイヤーの破壊は、ローエンド型破壊よりも限定的な意味を持つ。「既存企業に収益をもたらしてはいるが、過剰満足である顧客に対して従来の基準に沿った満足できる性能を低価格で提供するイノベーション」と定義されるローエンド型破壊では、どの程度の性能と価格の製品を提供すればよいかに関して、必ずしもはっきりしない。これに対して、未破壊のレイヤーを破壊するためには、既に破壊されたスタックの価格及び性能と、同程度の価格と性能を持つPFを提供すればよい。

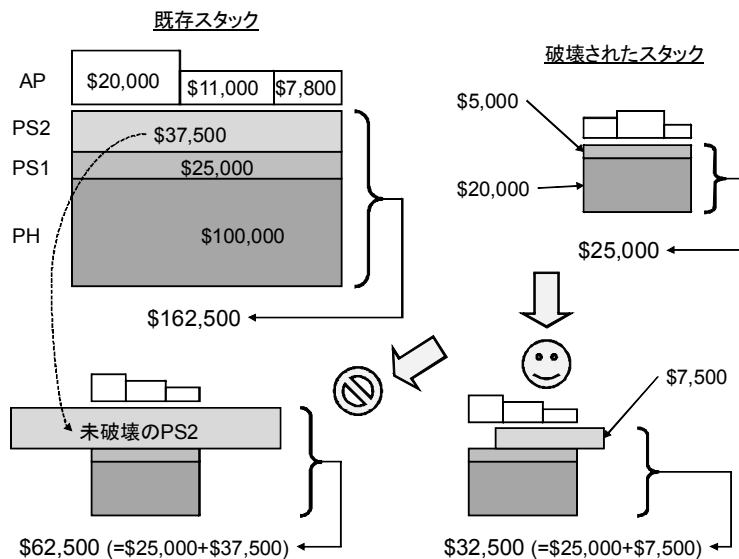


図6 未破壊のレイヤー

4.2 レイヤー優先度

レイヤー優先度は、「どのレイヤーの製品の売上をより優先するか」を表す。例えば、Microsoft 社はサーバーOSのWindows NTの売上を、DBのSQL Serverの売上よりも優先していたと考えられる。同社は、実際にできるかどうかは別にして、NT上のDBとしてSQL Serverのみが動作するようにすればSQL Serverの売上がより増加すると思われるにも関わらず、NT上で他のDBを動かすことに積極的には拒否反応を示さなかった。また、SQL Serverを他のOS上で動作させれば売上が増加する可能性が高いのに、SQL Serverを他のOSには対応させなかった。これも、NTの売上増加を優先したためと思われる。つまり、NTの売上増加にはなるが、SQL Serverの売上の減少につながる施策をとっていたと推定できるため、相対的に「OSの優先度を高く、DBの優先度を低く」設定していたと考えられる。

レイヤー優先度は、「自社製品を提供しているレイヤー」であるレイヤースコープ内で設定されるものである。例えば、Microsoft社は1993年以降、DBとしてSQL Serverを、OSとしてWindows NTを提供している。この場合、同社のレイヤースコープはDBとOSとなる³¹⁾。

レイヤー優先度が生じる理由は二つある。一つは、既存事業者は、できればレイヤースコープに含まれる複数のレイヤーの全ての売上を維持または増加させたいが、現実には売上あるいは利益額が小さい方のレイヤーよりも売上あるいは利益額が大きい方のレイヤーの売上あるいは利益額を維持または増加させることを優先させるためである。もう一つは、スタックの多様性を増すことで、複数レイヤーの売上または利益額を、全体として維持または増加させることを指向するためである。

あるPFソフトウェアを提供している企業が、その隣接上位レイヤーの補完製品となるPFソフトウェアも提供しているとき、他の企業が隣接上位レイヤーでPFソフトウェアを提供することに寛容であることは、実際には難しいであろう。PFソフトウェアは、多様な補完製品が動作することを前提に作られており、多くの補完製品が提供されるほどPFソフトウェアの価値は高まる。しかし、隣接上位レイヤーを自社のレイヤースコープに含めるのであれば、隣接上位レイヤーでも自社の製品が利用されることが望ましい。例えば、Microsoft社はNTとSQL Serverの双方の売上を最大化したいであろう。しかし、Oracleの製品が動作しなければNTの価値が減ってしまうため、SQL Serverの売上を最優先することはできないのである³²⁾。

5. 新規参入の成功要因

第3章での事例の分析結果から、第4章で設定した概念を用いて、「スタックを破壊すること」と「既存事業者と異なるレイヤー優先度を設定すること」の二つがデータベース市場への新規参入の成功要因となること本章では示す。5.1節では、新規参入に成功したOracle社とMicrosoft社の事例に共通する、二つの特徴を抽出する。5.2節では、MySQL社の新規参入ではOracle社とMicrosoft社の新規参入に共通する特徴の双方は見られなかったことを示す。5.3節では、「スタックの破壊」と「異なるレイヤー優先度の設定」の二つが、DB市場への新規参入の成功要因となると主張する。また、PFソフトウェア市場への参入一般の成功要因となる可能性を示す。5.4節では、先行研究との関係に触れる。

5.1 Oracle 社と Microsoft 社の新規参入

新規参入に成功した Oracle 社と Microsoft 社の事例に共通する、二つの特徴を抽出する。第一の特徴はスタックを破壊したこと、第二は既存事業者と異なるレイヤー優先度を設定していたことである。第一の特徴に着目したのは、階層性のある製品では、一部の PF を破壊するだけでは、利用者に全体的な利益をもたらせないからである。第二の特徴に着目したのは、同じレイヤー優先度を設定している場合は、当該市場において優勢である既存事業者が、相対的に大きい資金力を行使して、買収や価格競争を含む参入対抗策をとってくることを考慮するべきだからである。

Oracle 社は既存スタックを破壊した。具体的には、1978年に発売された100,000ドル程度の DEC のミニコン、VMS+VAX は、1,000,000ドル程度の IBM 社の AP+OS+HW のスタックを破壊した。Oracle 社は、この VMS+VAX 上で動作する、近年は50,000ドル程度のライセンス価格の RDBMS を1979年に提供した。また、Oracle の機能は IBM 社の DB である IMS よりも劣っていたが、VMS+VAX の発売によって AP+DB+OS+HW のスタックで未破壊となった DB レイヤーに自社の製品を対応させることで、VMS+VAX 発売の翌年には、AP+Oracle+VMS+VAX という破壊的スタックを提供した。

Microsoft 社も既存スタックを破壊して参入した企業である。OS/2上で動作する SQL Server のバージョン1.0が正式にリリースされたのは1989年だったが、1993年に、同年に発売されたサーバーOS の Windows NT 上で動作する、SQL Server V4.21をリリースした。Windows NT の発売によって、当時の50,000ドル程度のスタックの AP+Unix+ミッドレンジ・サーバー のスタックが、10,000ドル程度の AP+NT+IA サーバーのスタックへと置き換えられることもあった。その結果、DB が未破壊のレイヤーになると、NT+IA サーバーと同価格帯の、7,000ドル程度の RDBMS である SQL Server を発売して、DB レイヤーに低価格で参入し、既存の AP+Oracle+Unix+ミッドレンジ・サーバー のスタックを破壊した。ただし、後述するように、Oracle 社は、Oracle の標準エディションを投入することで、AP+Oracle+NT+IA サーバーのスタックを成立させた。

元々 Oracle 社は、既存事業者の IBM 社とは異なるレイヤー優先度を設定していた。IBM 社は、IMS (DB) と、MVS (OS)、System/370 (HW) を提供していた。IBM 社が1980年に RDBMS を OS に統合した System/38を販売したことから、同社が売上を優先していたのは OS と HW だとみなせる。DB の売上を優先するならば、IMS を他の OS や HW 上で動作させるはずである。Oracle 社が製品を提供していたのは DB のみであったため、当然 DB の優先度は高かった。Oracle は各種の Unix 上でも、Windows 上でも、IBM 社のメインフレーム上でも動作していた。つまり、既存事業者の優先度が低いレイヤーであった DB レイヤーに高い優先度を設定しての新規参入が、Oracle 社によって行われたのである。

その後の新規参入者であった Microsoft 社も、既存事業者の Oracle 社とは異なるレイヤー優先度を設定していた。IBM 社と Oracle 社の事例とは異なり、既存事業者の優先度が高い PF に対して、低い優先度を設定しての新規参入を成功させたのである。Microsoft 社のレイヤースコープは、DB と OS であり、優先度が高かったと推測されるのは OS である。DB の SQL Server の売上を優先させるのであれば、SQL Server を Windows 以外の OS 上でも動作させたはずだが、それはしなかった。Oracle 社

は DB が販売できればよいので、既存の AP+OS+HW のスタックが AP+NT+IA サーバーで置き換えられてもよかった。つまり、AP+Oracle+Unix+ミッドレンジ・サーバー のスタックは AP+SQL Server+NT+IA サーバーのスタックによる破壊を受けたが、Oracle 社は、Oracle の標準エディションを投入することで、AP+Oracle+NT+IA サーバーのスタックを成立させた。

5.2 MySQL 社の新規参入

MySQL 社の新規参入では、Oracle 社と Microsoft 社の事例に共通する二つの特徴の双方ともは見られなかった。ここでは、既存事業者を Oracle 社と Microsoft 社として、既存スタックを破壊していたか、異なるレイヤー優先度を設定していたか、の二点から、MySQL 社の新規参入を分析する。

MySQL 社が、その商用バージョンの最初の販売を成約したのは1997年であり、1998年に RDBMS 市場に本格的に参入して、Oracle を含むスタックを破壊した。AP+Unix+ミッドレンジ・サーバー のスタックを AP+Linux+PC のスタックが破壊した時点で、DB は未破壊の PF レイヤーであった。この時点では、Oracle は Linux+PC よりもずっと高い価格帯、標準エディションのライセンス価格でも 20,000ドル程度、で提供されていた。MySQL は、ライセンス価格は無料、サポート価格は1,000ドルという価格帯で提供された。ただし、MySQL は機能的には劣り、例えば Oracle が1992年に提供したストアドプロシージャとトリガーの機能を、MySQL が提供したのは2005年であった。

MySQL 社が、Windows 95と NT をサポートしたのは1998年だった (Windows NT 向けの SQL Server は1993年にリリースされていた)。しかし、MySQL 社は Microsoft 社製品を含むスタックを破壊したとは言えない。2,000ドル程度の AP+Linux+PC のスタックは、Microsoft 社製品が構成する 10,000ドル程度の AP+NT+IA サーバーのスタックというより、主として50,000ドル程度の AP+Unix+ミッドレンジ・サーバー のスタックを破壊した。また、当時の Windows NT は Unix ほど安定しておらず、IA サーバーの性能もミッドレンジ・サーバーに比べれば低かった。つまり、AP+NT+IA サーバーのスタックは、顧客に過剰満足をもたらしていたとは言えないという意味で破壊の対象になりにくかった。20,000ドル程度の AP+SQL Server+NT+IA サーバーのスタックが、情報系や基幹系システムに関与する人々に過剰満足をもたらしていたと考えるのは妥当ではないということである。

Oracle 社と MySQL 社は、DB に同じ優先度を設定していた。Oracle 社の DB の優先度は高かったが、MySQL は MySQL 社の唯一の製品であり、当然同社の DB レイヤーの優先度も高かった。それが、新規参入時に Oracle 社と Microsoft 社が、それぞれ既存事業者の IBM 社と Oracle 社からの熾烈な対抗策を取られなかったのに対し、MySQL 社は補完製品を提供する企業の買収や無料エディションの投入などの厳しい Oracle 社の対抗策に直面し、最終的には MySQL 社を買収した Sun 社ごと買収されてしまう要因になったと考えられる。

MySQL 社と Microsoft 社は、DB に異なる優先度を設定していたと考えられる。AP+MySQL+Linux+PC のスタックは、Microsoft 社にとっては AP+Linux+PC のスタックであった。同社がより強く懸念していたのは、AP+MySQL+Linux+PC のスタックが AP+SQL Server+NT+IA サーバーのスタックを破壊することではなく、AP+Linux+PC のスタックが AP+NT+IA サーバー (ま

たは PC) のスタックを破壊することであった。Microsoft 社が激しい対抗策を講じたのは、MySQL に対してではなく Linux に対してであった。一方、Oracle 社にとっては、AP+Windows NT+IA サーバーに代わる AP+Linux+PC の出現は好都合であった。しかし、SQL Server が NT とともに Microsoft 社から提供されていたのとは異なり、MySQL がなくとも Linux は Oracle の隣接下位 PF として利用できた。

5.3 データベース市場および PF ソフトウェア市場への新規参入の成功要因

以上の分析から、本稿は、「スタックを破壊すること」と「既存事業者と異なるレイヤー優先度を設定すること」の二つは、DB 市場への新規参入の成功要因となると主張する。Oracle 社と Microsoft 社の参入では、この二つの要因が含まれていた。一方、失敗した MySQL 社は、Oracle 社に対しては同じレイヤー優先度を設定し、Microsoft 社に対してはスタックを破壊してはいなかった。以下では、この二つの要因が DB 市場の主要な事業者の事例を説明するだけでなく、論理的にも一定の妥当性があることを示す。

仮に MySQL 社が Oracle 社と異なる優先度を設定していたら、新規参入は成功していただろうか。例えば、MySQL 社が MySQL と同価格帯の OS も提供していて、OS の売上を優先していたら、Oracle 社の MySQL 社への対抗策は厳しくはなかったのではないか。ただし、この場合には Microsoft 社が激しい対抗策をとった可能性がある。また、AP+SQL Server+Windows NT+IA サーバーのスタックが過剰満足を与えていたら、MySQL は SQL Server に対して優位に立ちえたのではないだろうか。

IBM 社製品を構成するスタックを破壊していなければ、Oracle 社は新規参入に失敗していただろう。Oracle が VAX/VMS よりもずっと高価だったとすれば、つまりメインフレーム上の DB を目指していたとすれば、新規参入は困難だっただろう。また、IBM 社が DB に高いレイヤー優先度を設定していたならば、IBM 社は Oracle 社に対して激しい反撃を仕掛けたのではないだろうか。

Oracle 社と同じレイヤー優先度を設定していたら、Microsoft 社は新規参入に失敗していただろうか。NT の売上よりも SQL Server の売上を優先したとして、SQL Server を Unix 上で動作させたとすれば、Oracle 社は SQL Server よりも安価なエディションを対抗製品として投入したのではないだろうか。Oracle 社は実際、MySQL 社に対抗するためには、無料エディションを迅速に投入したのである。

DB 市場への参入の成功要因は、PF ソフトウェアの参入の成功要因に一般化できるだろうか。第3章で示した事例では、DB のレイヤーだけでなく、HW と OS のレイヤーにおける参入も示した。DEC 社の VMS+VAX と Sun 社などによる Unix+ミッドレンジ・サーバー による参入は、基本的には HW のレイヤーへの参入だった。なぜなら、高性能な HW を売るために OS が必要だったと考えられるからである。OS のレイヤーに参入したと言えるのは、Windows NT と Linux である。

(サーバー) OS 市場への参入においても、スタックの破壊と異なるレイヤー優先度の設定は新規参入の成功要因になっている。NT も Linux もスタックを破壊したことが、**図4**と**図5**から見て取れる。そして、Unix+ミッドレンジ・サーバーを提供する企業が OS のレイヤー優先度を低く設定していた

ところに、Microsoft 社は OS の優先度を高くして NT を投入した。NT が含まれる同社の「サーバーとツール」セグメントの2007年度の売上高は約110億ドル、営業利益率は約33%であり、同社はサーバーOS 市場への参入に成功したとかがわける。このセグメントに含まれる SQL Server の売上高は約35億ドルであった。Linux は NT ではなく Unix を置き換えつつある。よって、本稿の成功要因は PF ソフトウェア一般の参入戦略成功の条件として有効である可能性がある。この一般化は、VMWare などの仮想マシンモニタや、WebLogic などのウェブアプリケーションサーバーなどの他の PF ソフトウェアの事例を研究することで、さらに一般性を担保していくものだろうと考えている。

5.4 先行研究との関係

スタックの破壊は、ローエンド型破壊 (Christensen and Raynor, 2003) よりも限定的な意味を持つ。スタックの破壊でも、あるレイヤーにおいてローエンド向けの低価格製品投入が行われる。この場合、単体の PF 製品として目指すべき価格と性能は、自社の製品が含まれると想定する「スタックの他のレイヤー」の価格と性能とのバランスによって与えられる。これに対して、しかし、「従来の基準に沿った機能を、低価格でローエンド顧客にとっては満足できる水準で」提供するローエンド型破壊では、直接競合となる「同種の既存製品」に対する価格と性能が問題にされる。

異なるレイヤー優先度を設定するという戦略は、プラットフォーム包囲 (Eisenmann *et al.*, 2007) のように支配的な PF を持つという前提を必要としない。第3章で述べた事例において、Oracle 社と MySQL 社は単一のレイヤーに参入していた。Microsoft 社は複数レイヤーに参入していたが、SQL Server が NT 上でしか動かなかったのはプラットフォーム包囲を狙ったものではない。どちらの製品も、新規参入製品だったからである。プラットフォーム包囲には、synergistic specificity (Schilling, 2000) を追求するという側面もあるが、SQL Server と Windows NT の組み合わせにおいては、開発リソースの節約と、NT の販売促進の意味合いが強かったと考えられる。

本稿における新規参入の成功要因には、ソフトウェアに関しては、変動費がほぼゼロであり (國領, 1999, Varian, 2000)、エディションを分けることで容易に性能の劣る安価な製品を作れるために、ローエンド向けの対抗製品を投入しやすいことが影響している。なお、本稿では、MySQL に関して、West (2003) に示されるようなオープンソース特有のビジネスモデルの問題とは捉えていない。一方、マルチホーミング・コストの観点から、レイヤー間の可視的な情報 (Baldwin and Clark, 1997) の影響についても、今後研究を進めていくべきだと考えている。

6. 結論

事例研究から、DB 市場への新規参入の成功要因は、「スタックの破壊」と「既存事業者とは異なるレイヤー優先度の設定」の二つだと分析した。ここで、スタックの破壊は「スタック全体としてローエンド型の破壊的イノベーションを起こすこと」である。また、レイヤー優先度は「どのレイヤーの売上をより優先するか」を表す。新規参入に成功した、つまり事業を黒字化し、かつ市場の寡占企業となりえた、Oracle 社と Microsoft 社の事例では、スタックの破壊と異なるレイヤー優先度の設定という二つの

条件を満たしていた。失敗事例として扱った MySQL 社の新規参入では、既存事業者の Oracle 社と同じレイヤー優先度を設定し、Microsoft 社に対してはスタックを破壊してはなかった。

本稿の貢献は次の四点である。第一に、PF ソフトウェア一般に拡張しうる新規参入の成功要因を特定したことである。第二に、PF ソフトウェア固有の特徴を考慮して、破壊的技術が出現しても成功している既存事業者 (Danneels, 2004) に関する知見を提示したことである。第三に、DB と、OS、HW の三つのレイヤーにおける、多企業間の破壊と競合の関係に関する事例を記述したことである。そして、第四に、破壊的イノベーションに基づいて、PF 製品の参入戦略の一般化への端緒を開いたことである。

注記：

- 1) 階層はモジュールの一種 (部分集合) である。各モジュールは設計の独立性を持つが、「動く」ためには別モジュールの存在が必要となる。階層においては、この依存関係が、上は下に依存するが、下は上に依存しないという意味で非対称となる。ソフトウェアは、階層性のある製品の代表的なものである。
- 2) 破壊的技術の概念は、破壊的イノベーションの概念 (Christensen and Raynor, 2003, Christensen *et al.*, 2004) へと拡張され、新市場型 (非消費者が、自分である特定の仕事を簡単にこなせるようにしてくれる、あるいはもっと便利に、特定の場所に向かなくても仕事がこなせるようにしてくれるイノベーション) とローエンド型 (既存企業に収益をもたらしてはいるが、過剰満足である顧客に対して従来の基準に沿った満足できる性能を低価格で提供するイノベーション) の二種類の破壊的イノベーションがあるとされる。
- 3) 二つの事業者間のシェアの比が 2 : 1 である状態で均衡しがちであり、比が 4 : 1 以上になると競争が困難になるとされる。
- 4) 公比 $(1/2)^2$ の等比数列の和は、初項を a^2 とすると $4a^2/3$ に収束する。よって HHI が 2500 に収束するのは、 $a \doteq 43.3$ の時である。そして、 $43.3 + 43.3/2 + 43.3/4 \doteq 75$ である。そして $43.3/4 \doteq 10$ である。
- 5) Oracle Corporation. "Form 10-K", <http://www.oracle.com>
- 6) Microsoft Corporation, "Form 10-K", <http://www.microsoft.com/investor/SEC/default.aspx>
- 7) Shankland, S. "MySQL hits \$50 million revenue, plans IPO", CNET News, April 25, 2007, <http://investor.com.com/MySQL+hits+50+million+revenue,+plans+IPO/2100-7344-3-6179290.html>
- 8) Oracle Corporation. (2009) Oracle Technology Global Price List. <http://www.oracle.com/corporate/pricing/technology-price-list.pdf>
- 9) Microsoft Corporation. SQL Server 2005 pricing. <http://www.microsoft.com/sqlserver/2005/en/us/pricing.aspx>
- 10) MySQL A.B. (2009/7参照), "MySQL TCO Savings Calculator," <http://www.mysql.com/tcosavings/>
- 11) Oracle Corporation. Oracle Software Technical Support Policies. <http://www.oracle.com/us/support/library/057419.pdf>
- 12) <https://www.microsoft.com/windowsserver2008/en/us/pricing.aspx>
- 13) IBM Power 780 Server Model 9179-MHB Using AIX Version 6.1 and DB2 9.7. http://www.tpc.org/results/FDR/TPCC/IBM_780cluster_20100816_FDR.pdf
- 14) <https://www.redhat.com/apps/store/server/>
- 15) IBM Corporation. System/370 Model 135. http://www-03.ibm.com/ibm/history/exhibits/mainframe/mainframe_PP3135.html
- 16) <http://ed-thelen.org/comp-hist/VAX-11-750.html>
- 17) <http://www.sparcproductdirectory.com/history.html>
- 18) Garry, C. (2003), MySQL: Open Source and the Commodity Effect. *MySQL White Paper*. Available at <http://www.mysql.com>
- 19) Oracle Corporation. (2009/5参照) Oracle Timeline. <http://www.oracle.com/timeline/index.html>
- 20) Oracle Corporation. Oracle Documentation. <http://www.oracle.com/technology/documentation/index.html>
- 21) Microsoft Corporation. SQL Server History. http://www.microsoft.com/sql/demos/sql_timeline/main.html

- 22) The Public Linux Archive. <http://www.kernel.org/pub/linux/kernel/>
- 23) Said, K. 2005. Business revamp to MySQL debuts. CNET News.
http://news.cnet.com/Business-revamp-to-MySQL-debuts/2110-7344_3-5911631.html?tag=lia:rcol
- 24) Oracle Corporation. 2005. Oracle Database 10g Express Edition: Free, as in Beer. *OTN TechBlog*.
http://blogs.oracle.com/otn/2005/10/oracle_database_10g_express_ed.html
- 25) MySQL A. B. Overview of MySQL Storage Engine Architecture. MySQL 5.1 Reference Manual, Section 13.4.
<http://dev.mysql.com/doc/refman/5.1/en/pluggable-storage-overview.html>
- 26) Oracle Corporation. 2005. Oracle Announces the Acquisition of Open Source Software Company, Innobase. Press release. http://www.oracle.com/corporate/press/2005_oct/inno.html
- 27) Oracle Corporation. 2006. Oracle Buys Open Source Software Company Sleepycat. Press release.
http://www.oracle.com/corporate/press/2006_feb/sleepycat.html
- 28) Shankland, S. 2006. Oracle tried to buy open-source MySQL. CNET News.
http://news.cnet.com/Oracle-tried-to-buy-open-source-MySQL/2100-7344_3-6040197.html?tag=mncol
- 29) Sun Microsystems. 2008. Sun Microsystems Announces Agreement to Acquire MySQL, Developer of the World's Most Popular Open Source Database. Press release.
<http://www.sun.com/aboutsun/pr/2008-01/sunflash.20080116.1.xml>
- 30) Oracle Corporation. Oracle Buys Sun. *Press release*. <http://www.oracle.com/us/corporate/press/018363>
- 31) 本稿の事例分析においては、自社製品を提供するレイヤーの全てを含める必要はない。DB レイヤーへの新規参入を分析するのに必要なレイヤーだけを、スコープとして含めれば十分だと考えられる。つまり、SQL Server の新規参入を考察するために、オフィススイート製品を考慮する必要はないと考えられる。
- 32) Paker and Van Alstyne (2005) は、サイドに優先度を設定して全体としての収益を最大化することについて研究しているが、この水平的な優先度に対して、レイヤー優先度は垂直的な優先度とみなせる。

<参考文献>

- Adner, R. (2002), "When Are Technologies Disruptive? A Demand-Based View of the Emergence of Competition," *Strategic Management Journal*, Vol. 23, Issue 8, pp. 667–688.
- Baldwin, C. Y. (2007), "Design Theory and Methods", *MiniConference with L'Ecole des Mines-Paris* (presentation slides). <<http://drfd.hbs.edu/fit/public/facultyInfo.do?facInfo=bio&facId=6417>>
- Baldwin, C. Y. and K. B. Clark (1997), "Managing in an Age of Modularity," *Harvard Business Review* 75, no. 7, pp. 84–93.
- Baldwin, C. Y. and C. J. Woodard (2008), "The Architecture of Platforms: A Unified View," *Harvard Business School Working Paper* 09-034.
- Bower, J. and C. M. Christensen (1995), "Disruptive technologies: Catching the wave," *Harvard Business Review*, 73, no. 1, pp. 43–53.
- Burgelman, R. A., S. Inkinen, and C. Wittig (2004), "MySQL Open Source Database in 2004," *Stanford Graduate School of Business Case*, SM124.
- Burgelman, R. A. and C. Wittig (2006), "MySQL Open Source Database in 2006," *Stanford Graduate School of Business Case*, SM124(B).
- Christensen, C. M. (1997), *The Innovator's Dilemma: When New Technologies Cause Great Firms to Fail*, Harvard Business School Press. (クレイトン・クリステンセン、玉田 俊平太監修、伊豆原 弓訳 (2001) 『イノベーションのジレンマ—技術革新が巨大企業を滅ぼすとき』、翔泳社)
- Christensen, C. M. and M. E. Raynor (2003), *The Innovator's Solution: Creating and Sustaining Successful Growth*, Harvard Business School Press. (クレイトン・クリステンセン、マイケル・レイナー、玉田 俊平太訳、櫻井 祐子訳 (2003) 『イノベーションへの解 収益ある成長に向けて』、翔泳社)
- Christensen, C. M., A. Scott, and E. Roth (2004), *Seeing What's Next: Using Theories of Innovation to Predict Industry Change*, Harvard Business School Press. (クレイトン・M・クリステンセン、スコット・D・アンソニー、エリック・A・ロス、宮本 喜一訳 (2005)、『明日は誰のものか イノベーションの最終解』、ランダムハウス講談社)
- Danneels, E. (2004), "Disruptive Technology Reconsidered: A Critique and Research Agenda," *Journal of Product Innovation Management*, Vol. 21, Issue 4, pp. 246–258.

- Delaney, K. (2000). *Inside Microsoft SQL Server 2000*, Microsoft Press. (カレン・デレーニ、株式会社クイック訳 (2001)、『アーキテクチャ徹底解説 Microsoft SQL Server 2000』、日経 BP ソフトプレス)
- Eisenmann, T. R. (2007), “Winner-Take-All in Networked Markets,” *Harvard Business School Note*, 806-131.
- Eisenmann, T. R., G. G. Parker, and M. W. Van Alstyne (2006), “Strategies for Two-Sided Markets,” *Harvard Business Review*, 84, no. 10, pp. 92–101.
- Eisenmann, T. R., G. G. Parker, and M. W. Van Alstyne (2007), “Platform Envelopment,” *Harvard Business School Working Paper*, No. 07-104. <<http://www.hbs.edu/research/pdf/07-104.pdf>>
- Govindarajan, V. and P. K. Kopalle (2006) “The Usefulness of Measuring Disruptiveness of Innovations Ex Post in Making Ex Ante Predictions,” *Journal of Product Innovation Management*, Vol. 23, Issue 1, pp. 12–18.
- Henderson, B. D. (1976), “The Rule of Three and Four”, <<http://www.bcg.com/documents/file13258.pdf>>
- Komo, D. (2005) “Understanding Database Pricing and Licensing,” <<http://download.microsoft.com/download/e/2/3/e2341d27-107f-4613-ad97-eb277b48241e/DBPricing.docx>>
- Olofson, C.W. (2007), “World-wide Relational Database Management Systems 2007 Vendor Shares”, <http://www.microsoft.com/sqlserver/2008/en/us/news-reviews.aspx>
- Parker, G. G. and M. W. Van Alstyne (2005), “Two-sided Network Effects: A Theory of Information Product Design,” *Management Science* 51(10), pp. 1494–1504.
- Patterson, D. A. and J. L. Hennessy (2004), *Computer Organization and Design, Third Edition: The Hardware/Software Interface, Third Edition*, Morgan Kaufmann. (デイビッド・A・パターソン、ジョン・L・ヘネシー (成田光彰訳) 『コンピュータの構成と設計 第3版、別冊 歴史展望』、pp. 25、図7、日経 BP 社、2007年5月)
- Schilling, M. A. (2000), “Toward a General Modular Systems Theory and its Application to Interfirm Product Modularity,” *Academy of Management Review*, Vol. 25, No. 2, pp. 312–334.
- Varian, H. (2000), “Markets for Information Good,” in *Monetary Policy in a World of Knowledge-Based Growth, Quality Change, and Uncertain Measurement*.
- West, J. (2003), “How open is open enough?: Melding proprietary and open source platform strategies,” *Research Policy*, Vol. 32, Issue 7, pp. 1259–1285.
- 加藤和彦 (2009) 「階層構造をもつコンピュータ・ソフトウェアにおけるプラットフォーム戦略としての階層介入施策の考察」『日本経営学会誌』、No. 23, pp. 75–86.
- 経済産業省 (2008) 『機械統計年報』
http://www.meti.go.jp/statistics/tyo/seidou/result/gaiyo/resourceData/03_kikai/nenpo/h2dcd2008k.pdf
- 公正取引委員会 (2009) 「企業結合審査に関する独占禁止法の運用指針」 <http://www.jftc.go.jp/ma/kigyo-gl.pdf>
- 國領二郎 (1999) 『オープン・アーキテクチャ戦略』ダイヤモンド社
- 根来龍之、加藤和彦 (2010) 「プラットフォーム間競争における技術「非」決定論のモデル—ソフトウェア製品における WTA のメカニズムと対抗戦略—」『早稲田国際経営研究』、No. 41, pp. 79–94.
- 根来龍之、釜池聡太 (2010) 「ソフトウェア製品の平行プラットフォーム市場固有の競争戦略」『早稲田大学 IT 戦略研究所ワーキングペーパー』、No. 34
- 根来龍之、高田晴彦 (2010) 「製品戦略論における出発点の吟味」『早稲田大学 IT 戦略研究所ワーキングペーパー』、No. 35

