

---

---

# 疑似オープン化（レガシー資産有効活用）について

キグナス石油（株）

---

## ■ 執筆者Profile ■



倉持 信哉

1985年 キグナス石油（株）入社  
業務部需給課  
1987年 東燃システムプラザ出向  
メインフレームによる開発担当  
1990年 キグナス石油（株）  
システム部システム推進課  
2007年 現在 同社  
経理部情報システムグループ課長

## ■ 論文要旨 ■

ユーザーへの情報の見せ方、入力のしやすさは情報力を左右する大きなファクターの一つであるが、その点でメインフレームはまさにレガシーシステムと言える。そのため同業他社の多くはオープンシステムへの移行を選択した。しかし、現システムのすべてをオープンシステムへ乗り換えるためには30億円以上の投資が見込まれた。

そこで私は入力をオープンシステム、幹の処理を現状のままメインフレーム、出力をオープンシステムというサンドイッチを作り、ユーザーの利便上はあたかも全てがオープンシステムであるように感じられる「疑似オープン化」を実現し、必要な機能のみ順次これに移行していくことはできないか、と考えた。理論的には可能だが、同様の構築実績が見あたらないという状況でのスタートであり、失敗のリスクを抱えながらのシステム構築となった。

## ■ 論文目次 ■

<b>1. はじめに</b> .....	《 3》
1. 1  当社の概要	
1. 2  当社システムの特徴	
<b>2. 旧システムの現状と問題点</b> .....	《 4》
2. 1  システムの現状	
2. 2  発注システムの問題点	
2. 3  実績データ分析の問題点	
2. 4  セキュリティの問題点	
2. 5  オープンシステム移行の問題点	
<b>3. 疑似オープン化の発想</b> .....	《 6》
3. 1  問題の整理	
3. 2  疑似オープン化構想	
<b>4. 新システムの構築</b> .....	《 8》
4. 1  連携機能の種類	
4. 1. 1  オンラインリアルタイム	
4. 1. 2  非同期処理	
4. 1. 3  ファイル転送	
4. 2  各システムの機能と構成	
4. 2. 1  オーダーシステム	
4. 2. 2  セキュリティシステム	
4. 2. 3  販売情報システム	
<b>5. 疑似オープン化の評価と今後の課題</b> .....	《 14》
5. 1  疑似オープン化の評価	
5. 2  今後の課題	
<b>5. 最後に</b> .....	《 15》

## ■ 図表一覧 ■

<b>図1</b> 旧システムの処理イメージ図 .....	《 3》
<b>図2</b> 疑似オープン化のイメージ図 .....	《 7》
<b>図3</b> 3つの連携タイプイメージ図 .....	《 9》
<b>図4</b> オーダーシステム概要図 .....	《 11》
<b>図5</b> オーダー画面のプルダウン .....	《 12》
<b>図6</b> 販売情報システムイメージ .....	《 13》

# 1. はじめに

## 1. 1 当社の概要

当社の業務は石油卸売業である。東燃ゼネラル石油（株）が原油を仕入れ、これを精製してガソリンなどの石油製品を製造する。当社はこれら製品を引き取り、系列のSSや工場などへ石油製品を安定的かつ安全に供給することを使命としている。

資本金は20億円。社員数約110名。系列SS618カ所。自社油槽所3カ所。その他契約または相互利用の基地が全国に点在する。売上高は4,900億円である。

## 1. 2 当社システムの特徴

当社の基幹システムは富士通のメインフレームで稼働しており、受注から請求書作成までをこのメインフレームで行っている。基幹システムにおける情報の原始データは特約店からのオーダーデータである。例えば、「ガソリン4KL」というオーダーがそれにあたる。そして、「オーダーデータ＝出荷実績データ」であり、そのデータが会計処理までつながっていくことが特徴としてあげられる。

疑似オープン化前のシステム（以後旧システムと表記する）による処理イメージを図1に示す。

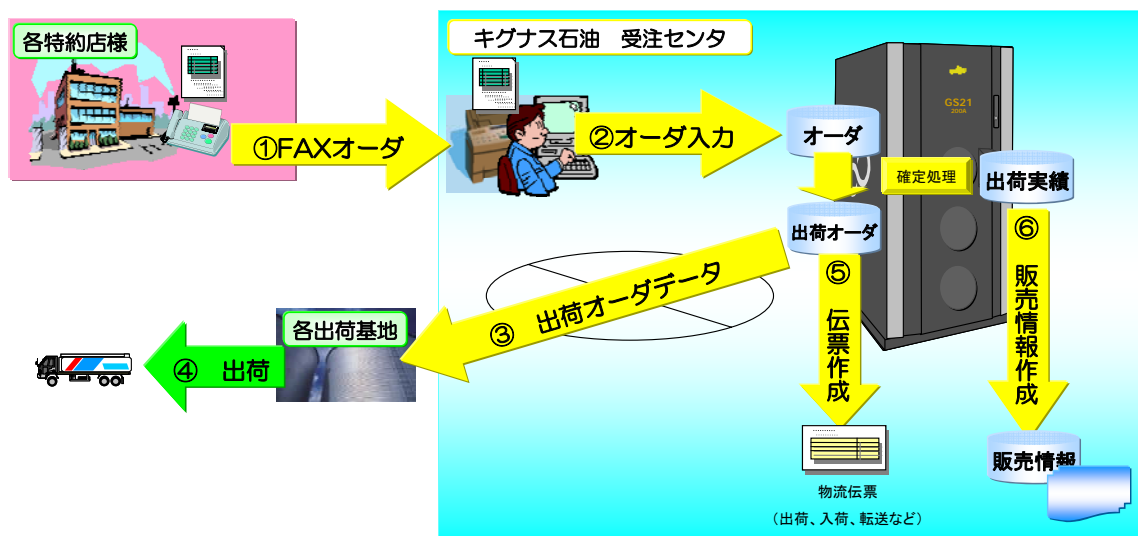


図1 旧システムの処理イメージ図

旧システムでのオーダーは、主にパソコン通信とFAXで受けていた。パソコン通信でのオーダーでは特約店が入力したデータそのものがオーダーデータとなるが、FAXの場合はその内容を受注センターの従業員がメインフレームの端末に転記入力することでオーダーデータが発生する。受注センターでこれに配送情報を加えて、出荷基地のコンピューターに転送すると、出荷基地では出荷が可能な状態となる。

配送情報とは、どのローリーがどこの出荷基地で積載し、どういう順番でSS（サービスステーション＝ガソリンスタンド）に配送していくかという情報である。配送情報の作成は配車システムと呼ばれる基幹システムとは別のシステムで作成しているが、この論文の中では大きな意味を持たないので、その説明と図への記入は割愛する。

配送情報が付加されたオーダーデータを元に出荷基地では出荷作業を行う。しかし、後に営業社員が販売分析に使用する出荷実績データや会計データは各出荷基地での出荷データから作成されるのではなく、元のオーダーデータから作成される。もしも何らかの事情で出荷されなかったオーダーがあったとしたら、それはオーダーデータにさかのぼってキャンセルする。したがって、「オーダーデータ＝出荷実績データ」となり、その後この出荷実績データを元に配送、在庫、仕入、売上等のデータが作成されるのである。これが、オーダーデータが基幹システムにおける原始データ、という意味である。

## **2. 旧システムの現状と問題点**

### **2. 1 システムの現状**

当社では 20 年以上にわたってオーダーメイドでメインフレームのシステムを構築しており、すでに 300 万ステップを超える資産を抱えている。これらすべてをオープンシステムに移行するには 30 億円を超える投資が必要と見込まれる。

旧システムの稼働はきわめて安定していてトラブルはほとんどなく、手のかからないシステムである。そのために運用スタッフも少人数で済み、トラブルによって社員の仕事を遅らせることもない。

さらに、システムの根本的な機能としては、なんら変える必要がないという現状があった。問題にされるのは入出力のユーザーインタフェースとセキュリティだけで、データを処理する中核の機能については問題なかった。それは、商品の種類、販売形態、流通機構などの変化が劇的には発生してこなかった石油業界だからこそだともいえる。問題については「2. 2 発注システムの問題点」以降で説明する。

### **2. 2 発注システムの問題点**

「1. 2 システムの特徴」でわかるとおり、当社にとって大事なデータの元はすべてオーダーデータにある。パソコン通信によるオーダーでは受信したデータそのものがオーダーデータとしてメインフレームに保存されるが、FAX オーダーの場合は特約店から送られてきた FAX を見て受注センターの従業員が画面入力する。そのため、打ち間違えなどによる誤りが生じる可能性がある。このデータを元に主要なすべてのデータが作成されるため、ここでの誤りは多方面に広がる。そうなってしまえば修正に手間がかかり、その修正がまた別の誤りを発生させるケースも多く見られた。

また、パソコン通信によるオーダー方式はそのためのソフトウェアを PC に埋め込む必要があるため、発注用 PC を当社より特約店に貸与していた。その保守も当社で行う必要があるが、陳腐化すれば PC 本体を買い換えなくてはならなかった。さらに、マスタ類を発注用 PC に保持しているが、その内容はメインフレームのマスタと同期をとる必要がある。そのため、発注用 PC は当社から新しいマスタ情報を通信で取得するのだが、当社のメインフレームと発注用 PC が通信で接続されるのはオーダーを行うときだけであるため、マスタの同期にはタイムラグが発生する。

このオーダー方式は大いに業務の効率化に寄与してはいたが、発注用 PC が複雑な機能をもつことと、マスタのタイムラグによりトラブルは絶えなかった。

この問題を解決するために、特約店に WEB 環境で利用可能な発注システムを開発する

ことを検討した。特約店にインターネットを利用できる PC さえあれば利用できるのも、パソコン通信方式の問題点をクリアすることができる。また、ここでは詳しく記載しないが、FAX オーダーを行っている特約店がいつまでも FAX にこだわる理由もほとんど解決できるため、基本的には FAX オーダーがなくなることになる。そうなれば、受注センタでの転記作業を割愛でき、誤りの防止と効率化を図ることが出来ると考えた。

これを実現するためにまずメインフレームの端末を WEB で特約店に提供する案を考えた。その理由は開発費の安さである。しかしそれでは機能制約が多いことと、当社の基幹業務システムを搭載するコンピューターに外部からアクセスさせることになり、それはセキュリティ上の懸念も大きいという理由でこの案を断念した。

つぎに、オープンシステムで発注システムを作り、ため込んだ受注データをメインフレームにファイル転送する方式を検討した。メインフレームとオープンシステムの連携は難しいとはいっても、ファイル転送方式なら多くの実績がある。しかしこの方式では発注した結果を特約店がその場で確認することが出来ないという問題があった。一見たいた問題ではないように思えるが、一度出したオーダーの結果を、時間をおいてから確認しなくてはならないというオペレーションにすれば、当社でのミスは減っても特約店でのミスが多くなる可能性があり、在庫切れなどのトラブルを生みかねない。また、オーダー時に必要なマスタ類の一部はメインフレームとダブルで保持するため、やはりここでもマスタの同期にタイムラグが発生する。さらに、特約店でのオーダー作業にかかる時間が長くなることで、特約店へのサービスの低下となることも問題視された。

こうした問題を防ぐため、どうしてもその場でオーダー内容のチェックを行う必要があった。しかし、オーダーをチェックし受注可否の判断を行うための情報のほとんどはメインフレームのシステムが抱えていてその内容は刻々と変化するので、この受注マシン単体では不可能な機能要求だと言えた。

### **2. 3 実績データ分析の問題点**

営業部門で実績データを分析する場面でも問題が発生していた。メインフレームが出力する情報の表現力はオープンシステムのそれと比べて明らかに劣る。画面での情報検索では一つの画面に表現できる情報量が少なく、無愛想に数字を並べることくらいしかできない。帳票においても同様で、情報出力というよりはデータリストのような帳票ばかりだった。

それを補うために、営業部門では分析資料はすべて Excel などの PC ソフトで作成していた。もともとはメインフレームでの画面出力もしくは帳票出力がその役目を担っていたのだが、PC ソフトの使いやすさ、分析能力の高さを見れば、そちらにシフトしてしまうのは自然の成り行きだった。

ところが、メインフレームと PC ソフトでは自由自在にデータの行き来が出来ない。そのため、メインフレームから出力した帳票をもとに、PC ソフトに転記する作業が発生していた。定型化した一部の資料についてはメインフレームのデータを PC ソフトに取り込めるようプログラムを作成して対応したが、操作性が悪く自由度もなく、根本的な解決には至らなかった。

## 2. 4 セキュリティの問題点

メインフレームで作成したソフトウェアにも当然、ソフトウェア毎にアクセスできる人物を限定できる機能が設けられている。しかし、20年以上前に作成されたこのセキュリティでは人ごとにアクセス制限をしているのではなく、所属部署と役職によってコントロールしていた。

ところが、社員の減少とともに一つの部門が担当する業務の範囲が以前に比べて広くなると、同じ部署に在籍していてもアクセスしても良いデータが異なるようになった。セキュリティ強化の必要性も強く謳われるようになった現在では、アクセス許可は最小限に抑えるべきだが、部署によるアクセス制限ではこれに対応することが出来ない。今求められているのは、一人一人細かくアクセス権限を絞り込むことなのだが、旧システムでのアクセス権限を変更するには、ほとんどすべてのプログラムを改訂する必要がある。しかしその作業があまりに膨大な費用を発生させることは目に見えていた。

## 2. 5 オープンシステム移行の問題点

上記3つの問題を解決するため、「オープン化」が言われるようになった。

基幹システムがオープン化し、発注システムもオープンシステムで構築し、そして販売情報システムもオープンシステムで新たに構築すればそれらの連携は容易であるため、要件にあった発注システムを特約店に提供できるばかりか、いままで出来なかった情報提供を特約店に対して行うなど、サービス向上も見込める。

実績データの分析についても画面をWEBで提供すれば、グラフや表を使うなど方法を豊かな表現力で提供することができる。帳票出力についてはExcelやPDFなどどんな形式でも提供が可能だろう。

オープン化をすればそのときにセキュリティについても見直すことができる。

同業他社を見ると、多くの会社でメインフレームからオープンシステムへの移行を実行していた。必要な経費はかけなくてはならないのだから、この選択肢はごく自然であるとも言えた。

しかし、情報システムにかかる経費について当社は大きな問題を抱えていた。それは「スケールデメリット」である。会社の規模が小さければデータは少なくなる。データが少なればハードウェアのスペックは落とせるが、ソフトウェアはそうならず、どうしてもスケールデメリットが発生する。単なる全システムのオープン化ではそのスケールデメリットをまともにうけてしまうことになるだろう。

それを考えると、「他社もそうだから」だけで全システムのオープン化を選択することは躊躇せざるを得ず、違うアイデアも模索することになった。

## 3. 疑似オープン化の発想

### 3. 1 問題の整理

今までにあげた事柄を整理すると、次のようになる。

- (1) 現在のシステムには大きな問題があるが、それはオープン化でクリアできる。
- (2) 同業他社はオープン化に向かっている。
- (3) しかし、全システムオープン化は他社に対してスケールデメリットがある。

### 3. 2 疑似オープン化構想

整理した問題を眺めたとき、もしも他社に並んで全システムのオープン化を行うことなく、オープン化で得られる利点だけをものにできたらすべてが解決する、という少々図々しい考えを持った。機能的な問題を全てクリアしつつ、全オープン化のような多大な投資をしない方法を見つけ出せばスケールデメリットも発生せず、情報システム部門として会社の競争力強化に貢献することが出来る。

いろいろな案や組み合わせを検討したが、最終的には意外と単純で理想的な案が残った。それは現状のメインフレームをそのまま残し、入力と出力だけをオープンシステムで開発し、その二つでメインフレームを挟み込んでサンドイッチ状にする構成である。

このときオープンシステムとメインフレームとはリアルタイムに連携をとることができなくてはならない。もちろん新しいセキュリティを組み込むことも忘れてはならない。

もしこれが実現できれば、最も重要な中核のシステムはメインフレームのままで、ユーザーから見ればあたかもすべてがオープン化されたように利用が可能である。中核部分を改訂せずに手足である入出力だけをオープン化するのだから、すべての入出力をこのシステムに切り替えても、開発費は3分の1程度に抑えられると試算した。しかも、必要な入出力だけに限って行えるので、さらにコストダウンを図ることができるはずだ。

これを私は「疑似オープン化」と名付けた。この構成イメージを図2に示す。

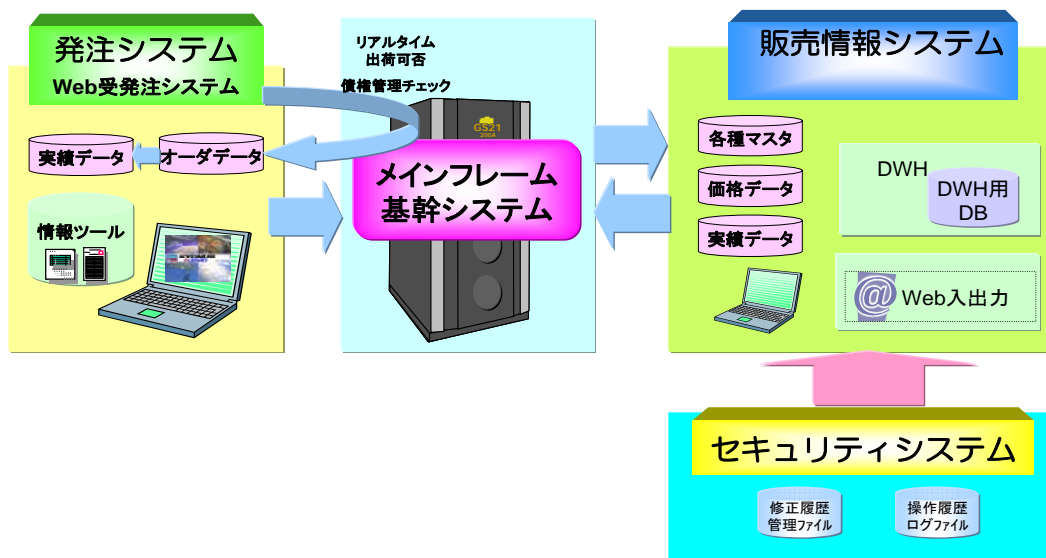


図2 疑似オープン化のイメージ図

具体的にはつぎのような構成ができればよい。

- (1) ユニックスサーバで発注システム、販売情報システム（入出力機能のみ）、セキュリティシステムを開発する。
- (2) メインフレームは基本的には現状のままとする。
- (3) 発注システム、販売情報システムをメインフレームの基幹業務と接続し連携させる。そして、販売情報システムは新しいセキュリティシステムの認証を受けての利用とする。

機能的には次の事をクリアする必要がある。

- (1) 発注システムでは特約店が WEB 画面で提供した発注システムで商品の注文を行えるようにする。注文した内容はリアルタイムで内容のチェックを受け、その場で受注の可否を表示できる。
- (2) 発注システムは単に商品の注文をするだけでなく、出荷実績表や商品のスペック表、請求書、マニュアル類、販促チラシなどが出力できる情報ツールとしての機能を持つ。これを実現するためにはメインフレームにある出荷実績情報、請求情報をはじめとする数多くのデータを条件によって収集し、発注システムにファイル転送してライブラリーに納める。
- (3) 営業社員向けの販売情報システムでもユーザーが WEB 画面を使って操作をする。要求した条件でメインフレーム内のデータを収集し、販売情報システムへ引き渡す。画面で提供するだけでなく、帳票出力も可能とする。
- (4) 販売情報システムにアクセスするためにはセキュリティシステムで認証を受ける。このセキュリティシステムでは一人一人に対してどの項目で何が出来るのかを事細かに設定できる。これによって最小限の権限を付与することが可能となる。

新たに作るセキュリティシステムはオープンシステムによる入出力に対してのみ作用する。入出力をオープンシステム側につくり、同システムのメインフレーム側の入出力機能を止めてしまえばセキュリティは確保できるので、特に強力なセキュリティを必要とする業務から順に移行するなどの工夫をすることが可能だ。

## **4. 新システムの構築**

### **4. 1 関係機能の種類**

疑似オープン化はあたかもユニックスサーバ同士を通信させて構築しているかのような動きを、メインフレームを挟んで行うことになる。このシステムを構築するためには、

1. 双方の同期をとるオンラインリアルタイム処理
2. 一方が停止していても、起動後にデータ更新を反映できる非同期処理
3. ファイルを一括で転送するファイル転送処理

の3種類のタイプの連携方法でメインフレームとユニックスサーバが稼働でき、さらに実用的な速度で連携を行う必要がある。3つの連携タイプを図3に示す。



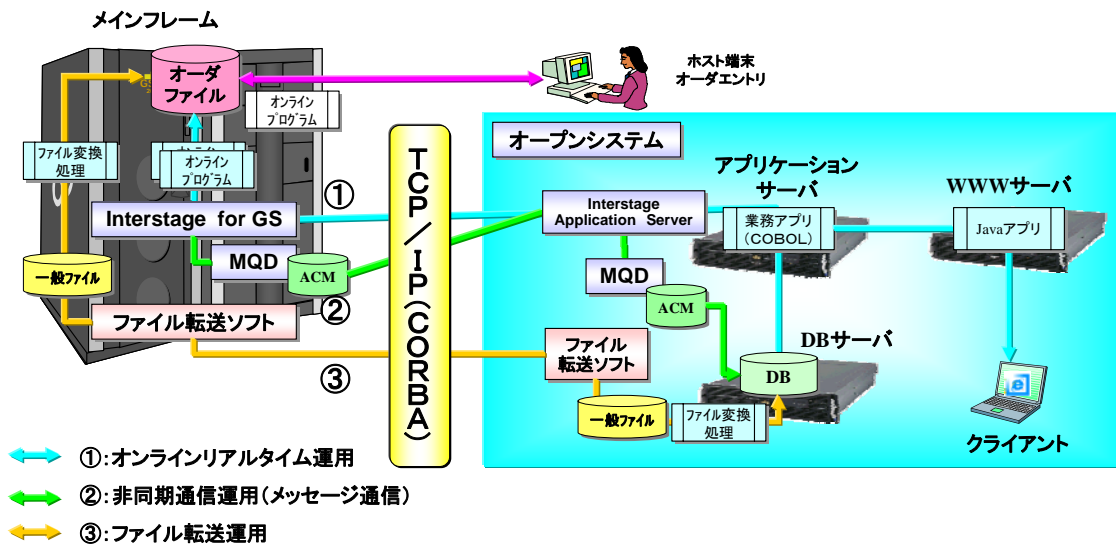


図3 3つの連携タイプイメージ図

連携の方法については上にあげたとおりだが、このシステム的设计思想として大事なことがもう一つある。それは、どのコンピューターが親なのか、ということだ。

このシステムを考案するにあたって柱になったのは、現在持っているメインフレームの完成度と安定度だ。それがあからこそこれを残し、手足とも言えるインタフェース部分のみをオープンシステムで構築しようと考えた。しかし、手足にもメインフレームと同じデータがなければ不便な場合が多々あり、同じデータを少々違う形で重複してオープンシステムに保持することがあるので、同期をとる必要が発生する。

このシステム構築案では親はメインフレームである。親がメインフレームであるということは、メインフレームから子供であるオープンシステムのデータの重要な部分はいつでも再生できるということである。これをしっかりと取り決めることは、データ保全につながる。

さて、それでは3つの連携方式について個別に説明をする。

#### 4. 1. 1 オンラインリアルタイム

リアルタイム処理はまさに疑似オープン化の要となる。この機能こそがオープンシステムとメインフレームとを一体化させ、疑似オープン化を果たす中心である。たとえば、受注の可否判断などに利用する。オーダーを受けたらその内容をメインフレームに問い合わせ、結果をもらうという作業を瞬時にこなさなくてはならない。

メインフレームでのチェック項目は商品在庫、その出荷基地の入構許可をされているローリーかなど多数有り、特約店への債権状況により算出される取引限度額をチェックしての取引可能な可否チェックまで行う。つまり、メインフレーム側ではオーダーを受けるときにいくつものマスタやデータファイルを検索して受注の可否をオーダーシステムに返すことになる。

メインフレームとユニックスサーバが同時に起動していることを前提に出来る仕組みにおいては、ほとんどをこの方式で対処できる。チェック後のオーダーデータ自体もこの処理でメインフレームに書き込む。

この方式でもっとも注意を払うことはレスポンスだ。Web 画面でいつまでも応答がないと、ユーザーは大変な不満を感じることになる。そういう待ち時間を作らないタイミングで応答を返したい。また、双方のデータに食い違いが発生するようなことは絶対にあってはならない。

この機能を実現するために種々のツールを検討したが、その結果、富士通の NETSTAGE に注目した。NETSTAGE はメインフレームのオンラインプログラムと同様にトランザクションレベルでの制御・構築を可能にし、かつ Web アプリケーションと AIM オンラインプログラムとの連携を CORBA 通信をとおして容易に図ることができるソフトウェアだ。これを使いデータ処理の信頼性とレスポンスが得られたのならば、疑似オープン化実現のキーとなるはずと考えた。

そこで当社のシステム構築を担当している大興電子通信（株）に相談したところ、「NETSTAGE での構築経験は複数あり構築をするための技術的な自信はある。しかし、今回ほど厳しい条件で NETSTAGE をフルに使ったことがないので、レスポンスの面で十分なスピードを得られないことも考えられる」と、含みのある回答だった。そこで、ユニックスサーバを大興電子通信（株）に持ち込み、大興電子通信（株）のメインフレームとの間でレスポンスのチェックを徹底的に行った。

その結果、まるでユニックスサーバだけで動作しているような錯覚を覚えるほど素早い連携を取ることが可能であり、信頼性も高いという結果が得られた。この時点で疑似オープン化は成功すると確信した。

#### **4. 1. 2 非同期処理**

非同期処理は主にマスタの受け渡しに利用する。発注システムとメインフレームの両方に必要なマスタは数多くあるが、これらは発注システムの稼働中には常に同じ内容になっていることが求められる。

例えば新しい納品先は毎日発生する。その時、メインフレームとオープンシステム双方のマスタを更新することは手順の無駄であるだけでなく、トラブルの元だ。そこで、メンテナンスはメインフレーム側だけで行うこととし、メインフレーム側のマスタが更新されるとオープンシステム側も更新されるようにしたい。

しかし、メインフレームの稼働はほぼ 24 時間に近いが、メインフレームの稼働中に必ずしもオープンシステムが稼働しているとは限らなかった。しかし、メインフレームの利用者がオープンシステムの稼働時間まで考慮して仕事をするのは効率的とは言えない。そこで、オープンシステムが停止していた場合には、オープンシステム稼働後に自動的にマスタを更新してくれるような動きにしたい。それを行うためには非同期処理が必要となる。

これには NETSTAGE の MQD 機能を利用することとしたが、この処理でも NETSTAGE はオンラインリアルタイム処理と同様、十分な働きをしてくれた。

マスタ類はすべてこの方式で同期をとることとした。

#### **4. 1. 3 ファイル転送**

ファイル転送は主に特約店への情報提供と営業部門に提供する販売情報システムに使用する。要求された条件に従ってメインフレームでデータを集め、ファイルを作ってオ

オープンシステムに転送する。

情報を引き出すためのレポート機能ならば、ほとんどこの方式で対応が可能だ。

## 4.2 各システムの機能と構成

3つの連携方式を組み合わせることで、メインフレームをオープンシステムでサンドイッチにすることは出来た。その組み合わせで大概のことは可能となる。

そこで、実際に搭載した機能とその内容について触れていく。

### 4.2.1 オーダーシステム

オーダーシステムは疑似オープン化の第一弾として着手した仕組みである。基本的にはオープンシステムで作ったWEB画面でのオーダー入力の結果をメインフレームに届けるのが役目であるが、オーダーデータの入り口というだけでなく、特約店への情報提供の窓口でもある。

図4にオーダーシステムの概要を表した。

一番左の枠が特約店のPCをイメージしたもの。まずはPCから発注システムにオーダーを入力する。「KIND」と書かれた部分がオーダーシステムの乗ったオープンシステムである。

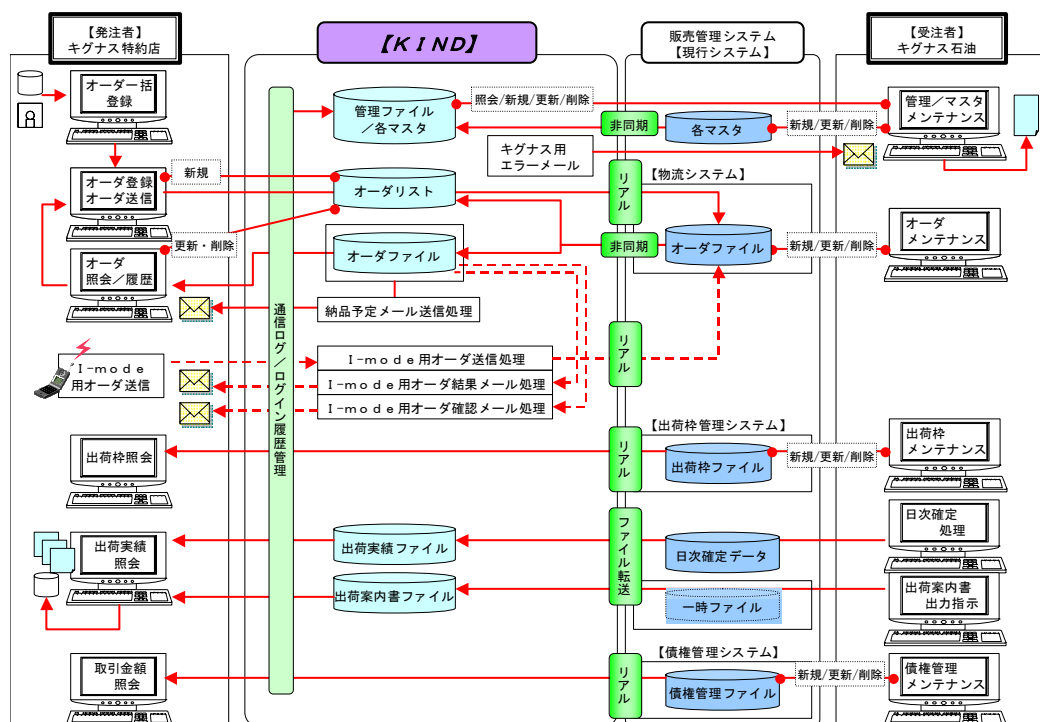


図4 オーダーシステム概要図

特約店からのオーダーは、ルールとして発注可能時間が決められているので、時間にはずメインフレームも起動しているという想定が出来る。その為、オープンシステムであるオーダーシステムからメインフレームへのデータの流れはオンラインリアルタイム方式が採用できる。

オーダーは一つずつメインフレームに送られるのではなく、オーダーシステムの中

に買い物かごを持つイメージで、複数のオーダーを登録することが出来る。すべてのオーダーを登録し終わった後に送信処理をすることにより、メインフレームに送られる。

メインフレーム側では種々のチェックを行い、一つずつのオーダーに対して結果を返していき、問題のなかったオーダーのみが受注される。

オーダー入力時の作業を簡単にするために、オーダー入力の WEB 画面ではなるべくプルダウンによる選択を採用した。商品、出荷基地などがそうだ。

図 5 に商品選択のプルダウンを採用したオーダー画面を示す。

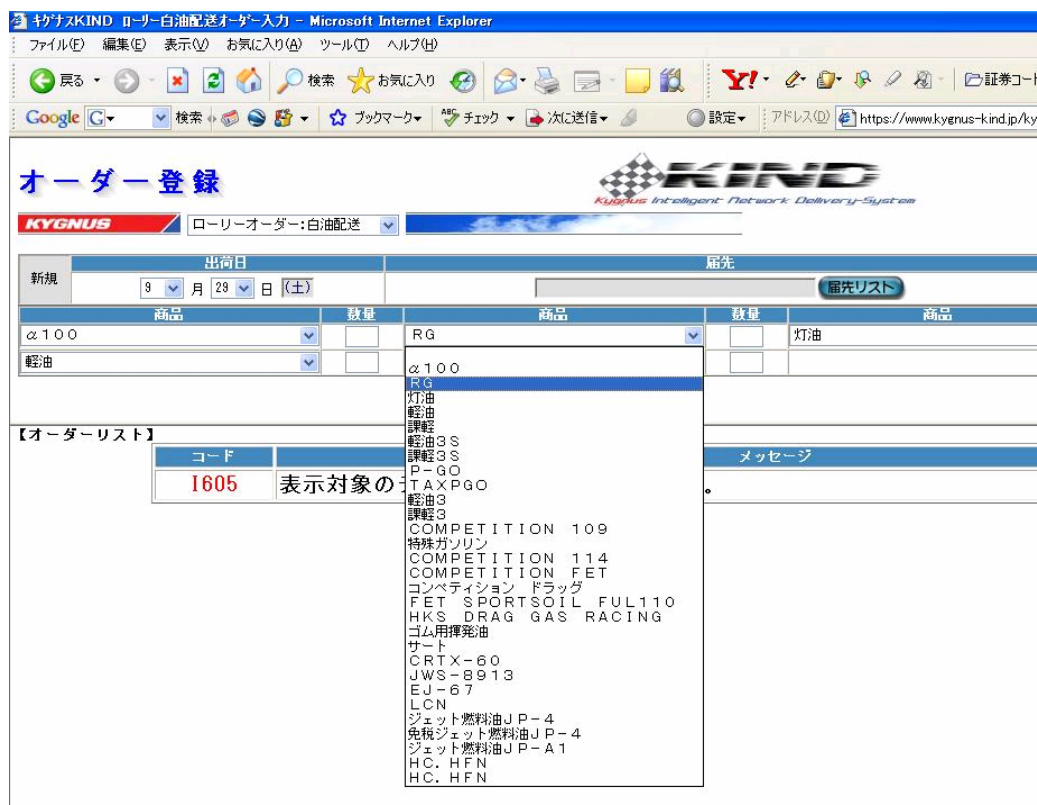


図 5 オーダー画面のプルダウン

プルダウンによる選択候補を表示するには、それらのデータを格納したマスタを発注システムに保持する必要がある。先に説明したとおり、メインフレーム側のマスタメンテナンス画面でマスタの変更を行うと発注システム側のマスタも更新される仕組みをとり、非同期処理を採用している。そのほかオープンシステム側のマスタをまるまる一気にメインフレーム側から転送し直す機能もあるが、これはファイル転送方式で行う。

図 4 でメインフレーム側の日次確定ファイル、一時ファイルからオーダーシステムの出荷実績ファイル、出荷案内ファイルを転送しているが、これは特約店が出荷実績の照会をした際に都度稼働する。ここに表記はないが、オーダーシステムでは PDF サーバで PDF ファイルを作成して特約店に提供する仕組みとしている。これらにはファイル転送処理で対処しており、このほかに請求書やスペック表、各種マニュアル等の提供も行う。

## 4. 2. 2 セキュリティシステム

セキュリティシステムはオープンシステムで作成され、オープンシステムで構築された販売情報システムのアクセスコントロールをしているので、直接的には疑似オープン化とは関係がない。しかし、ここで新たにアクセスコントロールが出来るようになったことで、メインフレーム側の全面的改修が不要となった。

アクセスコントロールを厳しく行う必要のあるデータへのアクセスプログラムについては、メインフレーム側でのレポート機能を削除し、レポート機能のみをオープンシステム側に作成する。そして、ファイル転送でオープンシステム側にデータを渡すことで簡単に新しいアクセスコントロールの元で情報の開示を行うよう改訂が可能である。

これについては今後徐々に対象を増やしていくことになる。

## 4. 2. 3 販売情報システム

図6に販売情報システムのイメージ図を示す。

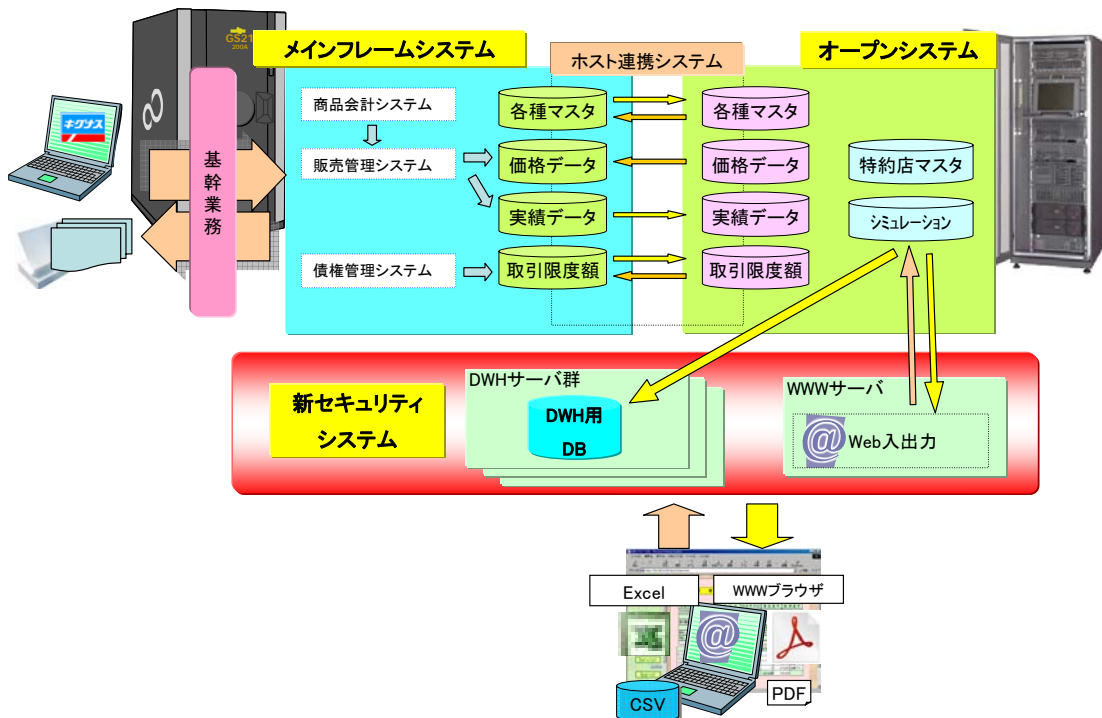


図6 販売情報システムイメージ

販売情報システムは当社の営業担当社員が利用する。まずはセキュリティシステムで認証を受け、アクセス権を取得する。

販売情報システムではその名前のおとり、販売活動の結果の情報を分析した資料を社員に提供するシステムだ。販売に関わるデータは基幹システムであるメインフレームで処理、保持されているので、オープンシステムで構築された販売情報システムはメインフレームと連携する必要がある。

現在のところ、マスタの更新はオーダーシステムと同様に非同期で連携し、販売情報はすべてファイル転送で処理されている。これは、当社の販売データの分析の最小時間単位が日であるためだ。つまり、前日の販売データを夜のうちにメインフレームより受け取っ

てしまえばよく、リアルタイムで受け取る必要性がないことと、販売情報システムは現在メインフレームからデータをもたらしているだけで、逆にオープンシステム側からメインフレームにデータの書き込みをすることがないからである。

図6の中に取引限度額というデータベースが描かれている。実はこれはまだ具体的に販売情報システムへの搭載が決まっておらず、開発には着手していないシステムである。取引限度額は特約店から得た担保と与信によって決まるため、随時発生する銀行からの入金や特約店からのオーダーで取引限度額の残高は頻繁に変化する。したがって、今後取引限度に関する情報の出口をメインフレームから販売情報システムへ移行するのだとしたら、ファイル転送処理では対応できない。さらに、取引限度額の変更などアクセス権限を厳重にする必要のあるシステムは、オープンシステム側に移行すべきだ。そうすると、オープンシステム側からメインフレーム側のデータを書き換えることになるが、先に述べたとおりこのシステムでは常にメインフレームが親である。したがってすべての基幹データの最新のものもメインフレームにあることが前提であるため、メインフレームへのデータのアップロードはリアルタイムでなくてはならない。よって、オンラインリアルタイム処理が必須になってくる。

現在、販売関連データはPDFファイルとExcelファイル、そしてWEBによる画面表示でユーザーに提供している。まだ、販売情報関連の出力の多くはメインフレームのシステムに残っているが、このシステムに移行したものについてはデータの転記作業は消滅したことになる。

最新のセキュリティで認証を受けて出力できるようになった販売情報はまだ一部であるが、その骨格となる機能は構築を完了した。

## 5. 疑似オープン化の評価と今後の課題

### 5.1 疑似オープンの評価

疑似オープン化は一回の開発プロジェクトではなく、システム構築の手法である。

2001年にその方向性を策定し、開発はその後徐々に進めてきた。2002年に発注システムを完成し、その後徐々に開発を進め、最近になってやっと「疑似オープン化」の評価が出来る段階までたどり着いた。

システムすべてを一気に疑似オープン化したわけではないので、全システムオープン化をした場合との正しいコスト比較はできないが、全オープン化を30億以上と見込み、疑似オープン化を全てのシステムに適用しても10億円以下のコストになると見込んでいる。しかも、疑似オープン化は一気にすべてを行う必要が無く、必要なところだけを徐々に行えるので、さらにコストダウンできるということになる。ここまでの開発はおよそ2億円の投資で行うことが出来た。

当初からオープンシステムだけで動く部分については何の心配もなかったが、オープンシステムとメインフレームでの連携部分、特にリアルタイムオンライン処理と非同期処理についてはレスポンスの悪さと障害の発生を懸念していた。しかし、レスポンスについては実験のときと変わりなく、一度たりともストレスを感じたことがない。オーダーシステムでタイムアウトによるエラーが何度か生じたが、それはすべて特約店のインターネット回線の問題であり、すぐに復旧した。また、連携における障害は驚くことに皆無であり、

NETSTAGE の完成度の高さを感じる。

入出力部分の開発はオープンシステムで行っているため、その設計の自由度は全オープン化と全く違いがない。システムの安定度については、安定した旧システムが中心になるため何ら問題がない。おそらくすべてのオープン化を図ったとしたら、数年間はトラブル対応に追われていたのではないかと想像できるので、この点も評価が出来る。

以上のことから、メインフレームで作成したシステムの根幹の部分を作り替える必要性があるのでなければ、疑似オープン化はきわめて有効な手段であると言える。

## 5. 2 今後の課題

今後の課題は、どの機能を疑似オープン化に乗せ、どの機能をそのままメインフレームで使うのかという切り分けである。疑似オープン化に乗せる理由はふたつある。一つはセキュリティ強化の必要性がある処理。そしてもう一つはオープンシステムでなくては十分な情報力を発揮できない処理である。これをどういう思想で、どういう基準で切り分けていくのかということが当面の課題になる。

いずれにしてもしっかりとした目的を持って、その目的を達成するための最善策を常に模索していきたい。

## 6. 最後に

今回の開発はいささか不安があった。情報システムに関する限り、カタログのスペックは必ずしも信用できない。単体では動いたとしても、他のシステムとの連携ではすべてを保証しているわけではないからだ。だからこそ、初期段階で徹底的にテストを行った。

しかし、このシステムの稼働により、疑似オープン化はすでに前例のある有効なシステム構築方法となった。もしも我々と同じくふくれあがった資産を捨てがたく思っておられるメインフレームユーザーがおられたら参考にしていただきたいと思います、今回の論文執筆に至った。

最後に、私の構想を技術的な側面からサポートし、数々の提案とチャレンジを繰り返しシステム構築全体の指揮を執ってくれた大興電子通信（株）の尾崎信哉次長、メインフレーム側の指揮を担当し、すばらしい連携を実現してくれた北友システム（株）の樋水常夫部長、そして両社のご協力と開発メンバーの熱意に、この場を借りて感謝したい。

以上