MIGARO. TECHNICAL REPORT

No.5 2012年秋

[創刊5周年記念号]















株式会社ミガロ.



		01
ごあいさつ		
Migaro T	echnical Award 2012 お客様受賞論文/ミガロ.テクニカルアワード	
【部門 1		
	JC/400 による取引先との Web-EDI システム構築	04
最慢 秀員	3 久保田 佳裕様●極東産機株式会社	
<u>~`</u> п [^#	Delphi と Excel を使用した帳票コストの削減	10
ゴールド賞	1 大久保 治高様●合鐵産業株式会社	
ゴニ !! . ! '	もっと見やすく、もっと使いやすい画面を 計 新谷 直正様●株式会社アダル	15
【部門 2		
優秀賞	- Delphi/400 で確認業務の効率化	20
投ノケテ	□ 荷田順 「娘●ハイドンノヤバン 「外氏云江	
優秀賞	取引 先申請システムでの稟議書作成ワークフロー 大崎 貴昭様 ● 森定興商株式会社	22
12707		
優秀賞	Delphi/400 で IBM i のストアードプロシージャを利用し、SQL 処理を高速化 ・ 島根 英行様●シルフ	28
創刊 5 周年記念	ミガロ .SE 座談会 お客様と共に歩む、お客様への熱い思い	31
Migaro.T	echnical Report 2012 ミガロ.SE 論文/ミガロ.テクニカルレポート	
Delphi/400 [初級者向け		38
Delphi/400 [中級者向け		50
Delphi/40([上級者向け		61
	DataSnap を使用した 3 層アプリケーション構築技法 尾崎 浩司●システム事業部 プロジェクト推進室	74
JC/400	JC/400 でポップアップウィンドウの制御 & 活用ノウハウ 清水 孝将/伊地知 聖貴●システム事業部 システム 1 課	86
Sガロ facebook ページ		

MIGARO. TECHNICAL REPORT

ごあいさつ

いつもミガロ.製品をご愛用いただき誠にありがとうございます。

さて、「ミガロ.製品をご利用中の技術者の皆様に、日々の開発に少しでもお役にたつような技術情報をご提供したい」という思いから 2008 年に創刊した『Migaro.Technical Report』は、このたび、節目となる第5号を発刊することができました。これもひとえに、ご多忙中にもかかわらず『Migaro.Technical Award (お客様論文)』にご寄稿いただいた多くのお客様、ならびに『Migaro.Technical Report』に対して貴重なご意見・ご要望をお寄せ下さったた皆様のご支援の賜物と、心より感謝をしております。

今回も従来と同様に、第 1 部は「Migaro.Technical Award 2012 お客様受賞論文」、第 2 部は「ミガロ. SE 論文」の 2 部構成としています。

第1部の「Migaro.Technical Award」とは、日々アプリケーションの開発・保守に携わるエンジニアの方々の努力と創意工夫の成果を顕彰することを目的とし、「Delphi/400」「JC/400」「Business4Mobile」などの弊社製品をご利用中のユーザー様を対象に実践レポート(論文)を公募し、厳正な審査・選考のうえ表彰する制度です。

今回は、『JC/400 により効率的に開発した Web EDI システム』や『Delphi/400 で用紙コスト削減と利便性向上を実現した帳票システム』など、創意工夫により業務課題を解決された論文をご寄稿いただきました。

また、『Migaro.Technical Award』では今回、従来のお客様論文にあたる「部門 1」に加え、「部門 2(My Tips 部門)」を新設いたしました。部門 2 は、「業務課題を解決した開発技術・テクニック」を簡潔にまとめていただくもので、今回は、『Excel と IBM i の情報マッチングによる業務効率化』や『トランザクション制御による SQL 処理高速化などのテクニック』などのノウハウをご紹介いただきました。

第2部「ミガロ. SE 論文」では、弊社 SE による技術論文を掲載しております。今回は、『Delphi/400でのスマートデバイス向け開発手法』や『Delphi/400インストーラでクライアントへのファイル配布手法』など、さまざまな応用テクニックを開発に活かしていただくための技術情報をご紹介いたします。

本レポートが少しでも皆様の開発・保守のお役に立てば幸いです。

最後に『Migaro.Technical Report』第5号を発刊するにあたりまして、多くのお客様・パートナー様にご支援、ご協力をいただきましたことを、この場をお借りして、あらためて厚く御礼を申し上げます。

2012 年秋

株式会社ミガロ. 代表取締役社長 上甲 將隆





お客様受賞論文/ミガロ. テクニカルアワード

JC/400による取引先とのWeb-EDIシステム構築 一RPG技術者1名 2カ月で開発し、本稼働を実現した手法

久保田 佳裕 様

極東産機株式会社 社長室 主幹技師



極東産機株式会社 http://www.kyokuto-sanki.co.jp/

昭和 23 年に畳の製造機器メーカーとして創業以来、職人さんの快適な職場環境作りと消費者の豊かな生活空間作りを2本柱として、伝統技術と先端技術の融合により、ユニークをおいインテリア施工省力機器はもとよりインテリア施工省力機器、カーテン縫製機器等、幅広く事業を拡大している。

1. Web-EDI 開発の 経緯

極東産機株式会社は、畳業界・インテリア施工業界のニーズに合わせたオリジナル製品を開発・生産し、全国の代理店経由で販売している。その出荷量は1日数千アイテム。全国の代理店から毎日、在庫の問い合わせがあり、関東・関西2カ所の受注センターで電話対応と注文の入力を行っている。

当社の在庫をリアルタイムで知りたい代理店のニーズと、その電話対応(在庫確認等)など受注センターの事務処理を軽減したい当社。そこで、双方にメリットのあるシステムを目標として、当社の在庫情報をインターネットを介して代理店に開示し、さらに情報照会だけでなく注文も行える仕組み「Web-EDI」の構築を決定した。そして、その開発ツールとしてJC/400を採用した。

2. 短期間での開発手法

2011年3月初めに、代理店向けWebサイトの構築を決定。それ以降は、4月に開発ツールの選定、5月に開発着手、11月に本稼働とする計画を立てた。しかし、3月11日に東日本大震災が起き、急きょ、震災対応システムの開発を5月~8月末に行うことになった。Web-EDIの構築はその後となったが、12月発刊の「総合カタログNo.12」号と同時公開としたため、開発期間が2カ月余りしかなく、短期構築の方法を模索した結果、次のような方法を取った。

(1) ミガロ. JC/400 基礎コースの受講

開発担当者は1名。RPGとNotesのスキルを持つものの、Web 開発の経験はない。

そこで、JC/400 の教育コース(所要 1日)の受講にあたって、事前に開発基 礎コースのテキストと JC/400 の開発マ ニュアルが入った CD-ROM をミガロ.か ら受領し、予習の上、講習にのぞむこと とした。

教育用のテキストは全95ページ。 JC/400の概要からHTMLのコーディング、RPGが自動生成された後の変更 箇所まで、未経験者にも分かりやすく解 説してある。このテキストをある程度理 解してから受講できたので、1日の受講 でJC/400 Designer の操作方法を含め、 一通りの開発手法をマスターできた。こ のテキストは、JC/400の開発に慣れた 今でも大事なバイブルとなっている。

(2) テンプレートの活用

Web サイトを一から開発したのでは 短期開発は困難である。そこで構築する サイトのイメージに合ったテンプレート を探し、それをカスタマイズするように した。

テンプレートは書籍、ネット上のフリーテンプレートから探した。また、ログイン画面から、ログイン後のメニュー、在庫照会画面、注文画面等の開発するすべての画面イメージをワープロで作り、その画面イメージを叩き台に営業部門と

図(概要図)

Web EDIシステムの概要

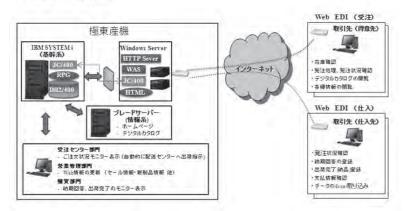


図1

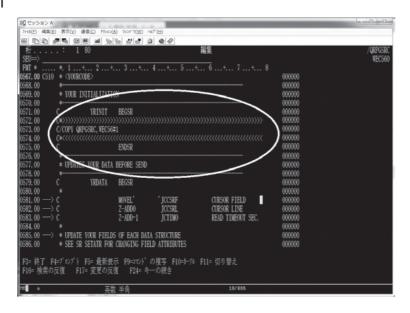
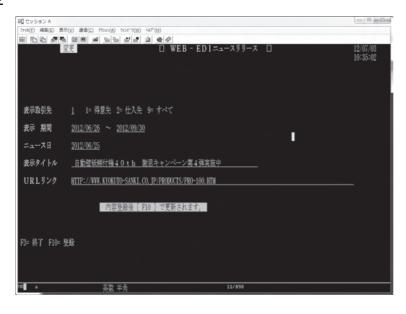


図2



打ち合わせをした。画像やアイコンも実際に使うものを利用し、完成後のイメージがしやすい資料を作成し、打ち合わせ時間の短縮を図った。

(3) ホームページビルダーの利用

Web 開発は、直感的な操作が可能なホームページビルダーを利用した。 Web 開発に慣れるまで使用していたが、 HTML等の記述方法が分かった後は、 テキストエディタでコーディングするようになった。

(4) HTML (画面) のパターン化

基幹システム (IBM i) の開発においては、既に基本的なプログラムパターンが完成しているので、一からプログラムを作ることはない。いくつかをプログラムパターン化し、それをカスタマイズし、再利用している(サブファイル型入力、非サブファイル型、照会、伝票印刷、メニュー等々)。Web 画面も同様に、いくつかの画面をパターン化し、利用することで開発工数を削減できた。

(5) RPG ソース自動生成

JC/400 は、Designer より RPG ソースが自動生成される。自動生成された RPG ソースは、ユーザーがコーディングできる場所に <YOURCODE> </YOURCODE> と記載されており、この間にロジックをコーディングする。多少の慣れは必要であるが、慣れてしまえばプログラムが標準化できるため非常に便利だ。また、開発生産性も日ごとに向上できた。

(6) COPY区とCALLの多用

COPY 区は、他のプログラムとの共通ロジックとして使うことが多いが、 <YOURCODE></YOURCODE> の間に COPY 区で記述することで、各プログラムで共通利用することはもちろんのこと、生成されたソースを見やすくすることができた。

また、画面が次々と展開していくプログラムも、RPGであれば1本のプログラムで作成できるが、JC/400の場合は画面展開の形によってはプログラムを分ける必要がある。その際に COPY 区を使うことで、パーツ化し共通利用できるので開発工数が短縮できる。

さらに、複雑なロジックは従来通り作り慣れている RPG、ILE-RPG で開発し、CALLで実行、実行結果のみを JC/400 のブログラムに戻してやることで、自動生成されたソースを意識することなく開発ができた。【図 1】

(7) JC/400 標準コンポーネントの利用

日付選択などは、JC/400 標準の日付コンポーネントを利用することで、開発時間を短縮でき、それと同時に、見栄えと操作性もよくなった。

(8) 他部門との協力連携

「利用規約」「セキュリティーポリシー」については、システム部門で原案を考え、管理本部に協力してもらい作成した。また、代理店へのシステムの説明では、営業部門を対象に事前に説明会を開催し、営業部門から各代理店に説明してもらい、順次申し込みをいただいた。

(9) ミガロ、のテクニカルサポートの活用

ミガロ.には、電話およびメールによるテクニカルサポートがある。開発当初は毎日のように電話し、サポートを受けた。即答いただけることが多かったので、開発時間の短縮を図ることができた。メールでは伝えづらいことも、電話で聞いていただけるので、JC/400を安心して開発できる、非常に有益なサービスである。

3. 開発の工夫

(1) 基幹システムから Web サイトが保 守できるメニューを開発

Web サイトのトップページの情報は、基幹システム側で動的に変更できるようにした。稼働後の保守は、できるだけシステム担当の負担にならないよう、営業担当者が、5250 画面を使ってユーザー登録や新着情報の登録・保守を行えるようにした。おすすめ商品の登録などは、リンク設定も簡単に行える。【図2】また、メニュー画面は Web-EDI の画面と同じ画面配置にし、管理しやすくした。【図3】(【図4】公開した Web-EDI メイン画面)。

(2) デジタルカタログとのリンク

12月に発刊された「総合カタログ」は、 Web からデジタルカタログとしても閲 覧できる。Web-EDI上で商品名をクリックすると、その商品のデジタルカタログのページにリンクできるようにした。これは、JC/400のRPGロジックのみで作成できた。【図5】

(3) 注文機能の表示・非表示

代理店によっては、専用 EDI のほうの「注文機能」を使うので、Web-EDIの「ご注文」ボタンを間違えて押さないよう非表示にしてほしいという要望があった。そこで、ユーザー登録の際に「注文機能を使う、使わない」を登録するようにし、「使う」の場合だけ「ご注文へ進む」ボタンを表示するように、ユーザーごとの機能制御を行った。

(4) カーソル制御

在庫照会画面では、商品コード(全品7桁)の7桁目が入力されると自動的に品名と在庫情報、出荷場所などを画面表示し、カーソルは次の商品コード欄に移動する。これも作業者の操作性をよくさせるためで、JC/400 Designer の「Autosubmit」の機能とRPG ロジックだけでカーソルを制御した。

(5) 在庫有無のロジック

在庫は数量ではなく、ホテル予約の空き室表示と同じように「○(あり)」「▲(少ない)」「×(なし)」で表示する。この定義の○と▲の設定が非常に重要で、ロジックは現在の在庫情報と直近の販売実績から基準を決め、自動計算し在庫判定する。JC/400は、基幹システムのデータを使えるので、メンテナンスは一切不要である。Web専用のデータベースを作る必要もない。

4. 代理店と社内の評価

「Web-EDI」サイトは、2011年11月に β 版として試験運用を開始し、12月にカタログ発刊と同時に正式公開した。そして、4カ月で200社を超える申し込みをいただき、「電話しなくても Webで在庫確認できる、また在庫確認した商品をそのまま注文もできるので便利になった」「注文した商品の出荷状況が確認できる」「在庫間い合わせの電話対応が減った(社内)」と代理店、社内からも高い評価をいただいている。

図3

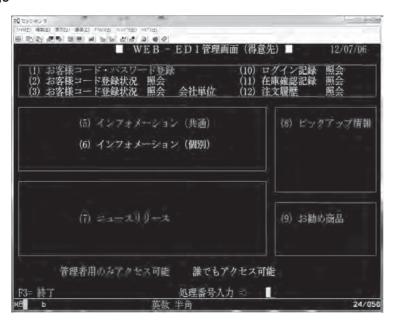


図4



図5



5. 今後 (運用後) の 取り組みについて

- (1) 発注データの CSV ダウンロード機能や Word による伝票発行などの機能を追加したサイトを新たに公開した (2012 年 5 月に稼働開始)。
- (2) SmartPad4i による生産現場でのタ ブレット活用や営業効率を高める 仕組みを構築中である。
- (3) 2004年にWebFacingで作成した 畳店とのネットワークも、JC/400 でリニューアルし、2012年8月より稼働を開始した。WebFacingは 廃止とした。(【図 6】はWebFacing、 【図 7】は JC/400 による画面)

JC400(SmartPad4i を 含 め) は、 IBM i の基幹システムを短期間で Web 化でき、社内だけでなく取引先も含めて 双方の事務作業の軽減が図れるので、今 後もさまざまなシステム構築を図ってい きたい。

 \mathbf{M}

極東産機株式会社 (詳細) http://www.kyokuto-sanki.co.jp/

「豊かな生活空間づくり、快適な職場環境づくり」を企業テーマとして、職人さんが手で行ってきた仕事の省力化、自動化に取り組んでいる。畳業界、インテリア内装業界の職人技を、新技術を駆使した各種機械で置き換え、伝統技術と先端技術の融合を図っている。畳機械メーカーとして創業以来、関連分野であるインテリア施行業界へと多角化を進め、メカトロ技術、エレクトロニクス技術を駆使したオリジナル機械の開発を進めると共に、コンピュータのソフト開発、販売まで手がける。顧客に喜んでもらえる製品を提供し、業界の近代化、さらには産業界、社会への貢献を目指している。

図6



図7



ゴールド賞

DelphiとExcelを使用した帳票コストの削減ーデザイン帳票ドット印刷からA4用紙レーザー印刷へ

大久保 治高 様

合鐵産業株式会社 総務部システム



合鐵産業株式会社 http://www.go-sun.co.jp/

70年余りの歴史を持つ鉄鋼メーカー「合同製鐵(東証1部)」の販社として設立。 鉄鋼材の専門商社として事業を営

鉄鋼材の専門商社として事業を営む。特に、建築用の鋼材に関しては 原料仕入から仕上げまで一貫して自 社で加工を行い、製品の付加価値を 高めている。

帳票発行システムの リプレース検討

合鐵産業株式会社は、鉄鋼材の専門商社として事業を営む。特に建築用の鋼材に関しては原料仕入から仕上げまで一貫して自社で加工を行い、製品の付加価値を高めている。また、老朽化した機械などから回収される鉄スクラップを販売。これが再び鋼材として再生されることで、資源保護にも貢献している。

2010 年に Delphi/400 を導入し、ミガロ.の協力により「受発注管理システム」の再構築を行い、大きな成果を上げた。

本稿では、前述とは別に IBM i の 5250 で運用していた「帳票発行システム」のリプレースを行い、コスト削減に大きな成果を出し、また業務改善を行った自社開発の事例について紹介する。

帳票発行システムとは、主に取引先に送付する、「納品書」「受領書」「請求書」の発行を行うシステムを指す。5250より処理を行い、ドットプリンタにて印刷を行う形式である。用紙については、複

写式の専用のデザイン帳票を業者に発注している。中でも納品書は6枚綴りで1枚の単価が非常に高いことから、Delphi/400導入以前から何度も課題として挙げられていた内容であった。

課題が解決に至らなかった要因としては、デザイン帳票をやめるとしても何らかの印刷手段が必要となる、またレーザープリンタ用の印刷ソフト、そのカスタマイズとそれらを用意する初期費用に対する効果が薄いということがあった。

Delphi/400の開発も1年が経過してきた時、自社で開発できることが少なからず見えてきた。複雑な入力系システムの開発については、やはり外注という選択肢をとらざるを得ないが、データのExcelへの出力などの簡易的な内容はできるようになっていた。

このことから、データの Excel 出力が可能なら、帳票の出力を自社開発するという検討に至り、2011 年秋より開発に着手、2012 年 1 月より運用の開始を行ったのが「Delphi の帳票発行システム」である。

開発要件

リリースした「Delphi の帳票発行システム」の要件としては、以下が挙げられる。

- 1. 印刷コストの削減
- 2. 5250 の機能維持
- 3. 拡張性 (後続の開発の容易性)

以上の内容が要件であり、Delphi/400 と Excel を使用することにより実現し た。順を追って説明する。

1. 印刷コストの削減

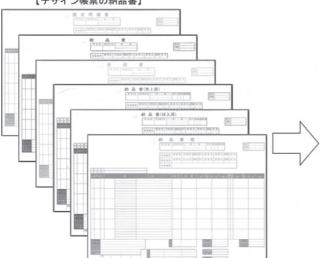
コスト削減は最も重要な要件であり、 コスト削減の成功がなければリプレース の意味がなくなってしまう内容である。 【図1】

以下のことから、コスト削減を行った。 (1) 用紙

印刷方法を Delphi と Excel の連携に よる A4 用紙のレーザープリンタ印刷と し、デザイン帳票廃止によるコストの削

図1 デザイン帳票の納品書とExcel台紙の納品書

【デザイン帳票の納品書】



【Excel 台紙の納品書】



図2 伝票出力制御画面



図3 伝票出力指示画面



減を行った。

納品書の台紙を Excel で作成し、発行する際に Delphi で台紙のコピーにデータを貼り付けて出力を行う方式とした。

そして、この処理を自動化することにより A4 コピー用紙での発行を可能とした。

(2) 出力枚数の調整

納品書の6枚綴りを、業務面で調整して3枚綴りとし、出力枚数によるコスト 削減を行った。

2. 5250 の機能維持

帳票の発行を単純に Delphi へ移すことの中にも、いくつかの問題があった。

(1) 画面の占有の問題

帳票を発行する際に、印刷指示を行った端末が印刷処理を行ってしまう場合、印刷している間(発行データをプリンタに送るまでの間)画面が硬直してしまう。 【図 2】

この問題については、発行指示をする 端末側では、印刷処理を行わないことと した。

対応策

・プリンタセッションの配置

5250と同様にプリンタセッションを 作成する。この端末で Delphi と Excel の連携を行い印刷処理を行うこととし、 発行者の端末に発行による処理の影響を 与えないようにした。

(2) 発行時間の問題

プリンタセッションは、発行処理をシーケンシャルに行うため、仮に全ての拠点を本社の1端末で処理することは可能であるが、印刷が集中してしまった場合はやはり時間がかかってしまう。今回の納品書は3枚綴りのため、3倍の出力が必要となる。つまり1拠点が100件を伝票発行したら、次の発行者は300件の伝票作成分の時間を待たないといけなくなってしまうのである。【図3】

対応策

・プリンタセッションの拠点配置

伝票を発行するプリンタセッションを 拠点ごとに配置し、発行元を分散させる ことにより問題を解決した。

発行指示データを作成する際に、ユーザーのサブシステムのジョブ(IPアドレスのセグメント)を指示データ(指示データは全社共通のファイル)に落とすことで、どこの拠点の印刷指示かを判断させた。これにより拠点ごとの印刷処理を可能にした。

(3) スプール処理

5250のスプール処理のように印刷を一時的に保留し、発行したいときに発行できるようにしたいとの要望が強くあった。実際に出力する場合、間違った操作をすることもあり、出力対象のスプールの中身を確認してから出力するという手順が業務として好ましかった。【図 4】

対応策

①保留操作

発行指示を行っても保留状態とし、開放を行わない限り印刷を開始しないようにした。

②伝票単位の出力制御

絞込みにより抽出された伝票ナンバーを画面上で確認でき、スプールの開放を行う前に伝票単位に出力を制御できる工夫を行った。

いらない伝票を出力してしまい出力後に捨てるという用紙の無駄を削減した。

3. 拡張性 (後続の開発の容易性)

納品書のリプレース後に、他のデザイン帳票にも同じ仕組みを適用していく計画である。開発工数をできる限り少なくするよう、5250では帳票単位に合った出力指示画面を1つに統一した。

また、納品書、請求書、売買契約書などの出力帳票に使用するワークファイルを同一のファイルで処理させ、開発の工数を削減する工夫を行った。

システム全体の仕組み

システムの全体像は以下である。【図5】

1. 伝票出力指示画面

クライアントのメニューより呼び出す ことができる。5250 でバラバラになっ ていた帳票出力の画面を1つに統一する とともにデータの抽出条件についてもま とめた。

(1) 条件設定と出力指示

出力対象の帳票を指定、絞込み条件を 入力して出力指示を行う。

(2) データ作成

IBM i 側で指示内容から出力指示データを作成する。

(3) 開放指示

対象を画面に表示し開放指示を行う (伝票単位の出力コントロールが行え る)。

2. 伝票出力制御画面

各拠点に1台設置されている5250のプリンタセッションの端末で稼働させた(今回は、発行を必要とする5拠点で稼働させた)。動作としては(1)ポーリング処理と(2)出力処理を交互に行う。

(1) ポーリング処理

- ・タイマーイベントでデータの存在確認を行う。
- ・対象のデータが存在している場合に 出力処理に遷移する。ない場合は帳 票区分(納品書、請求書、売買契約 書)をローテーションさせる。 (Delphiを IBM i に接続した時、 端末は IBM i サブシステム/ジョ ブの IP アドレスで識別されている。 これを利用し、どこの拠点のデータ かを識別する方法に利用した)

(2) 出力処理

出力指示データ内の帳票区分に沿った 処理を選んで処理を行う。

- ・IBM i 側で1伝票のデータを作成する。
- ・対象の帳票区分の Excel のテンプ レートをコピーしデータを貼り付け る。
- ・Excelをプリンタに出力する。

上記の処理を指示データのキーでファイルが読み取れなくなるまで繰り返し、 読み取れなくなった時にポーリング処理 に戻る。

評価

月間に出力する納品書枚数は約6000 枚あり、非常に稼働の多いシステムと

なっている。

最も重要な課題であったコスト削減については、デザイン帳票である納品書の全てが A4 用紙のレーザープリンタで出力できたことにより大きな成果を上げた。

業務面では、連票の切り離し、複写の 分解、ドットプリンタへの帳票の設置や 入れ替えというデザイン帳票独自の作業 がなくなり業務効率を上げることができ た。

また納品書の見た目として、取引先から見やすくなったという声もいただいている。

機能面としては、スプール中の伝票内容を確認でき、伝票単位に出力を制御できるところが5250にはなかった機能で使い勝手の向上が評価された。

開発面では、台紙がExcelということでリリース後の細かな調整に対して非常に柔軟に対応できたこと。また、納品書のみでなく他の帳票への適用を想定して拡張性を与えて構成したこともあり、納品書以降の帳票のリプレースが各デザイン伝票単位で2週間程度で完成できたことは評価できる内容であった。

今 後

5250の画面や印刷物では表現できなかったものを、Delphi/400では実現できる。この表現力を使って現行の5250のデータの照会画面やデータ出力のシステムをリプレースできれば、とても有意義なシステムになると考える。今回は対応を行わなかった帳票に加え、照会画面のリプレースが進行中である。

今後もユーザーの要望に応えつつ、さらなる利便性の向上や業務効率化のできるシステム開発に取り組んでいきたいと考えている。

M

ゴールド賞

もっと見やすく、もっと使いやすい画面を一在庫照会画面をDelphi/400でGUI化する

新谷 直正 様

株式会社アダル 電算室



株式会社アダル http://www.adal.co.ip/

業務用イス・テーブル・什器などの 製造卸販売、インテリア資材販売お よび設計・施工。 特注家具の製造・販売・施工を得意 とする会社である。

最初は好評だった緑色の在庫照会画面

「倉庫に電話をして在庫確認」が当たり前だった当社が、RPGで在庫照会のシステムを構築したのは2004年のことだった。倉庫担当者が不在の時など、営業マンへの情報の遅延イコールお客様への回答の遅延となるため、キーボードを叩けば在庫状況が見えるという、ある意味普通のことが、当時としては画期的なことに思えた。【図1】

噴出する社内の不満を 解消するためには

AS/400 での在庫照会が浸透すると、次の課題は在庫数の正確性、更新の迅速性、端末操作性の改良・改善であった。いくらパソコンで在庫照会ができるようになっても、画面に写し出される数字が正確でなければ意味がないし、加算・減算のタイムリーさも重要である。また、5250 画面の操作性においても AS/400

に不慣れな社員からは不満の声があがった。

5250 画面との決別、 Delphi/400 を採用

在庫照会に関連する全ての入力を5250画面で行っていた当時は、手書きやExcelで作成した伝票をオペレーターがまとめて入力するという方式をとっていたため、タイムリー性に欠けることは否めず、かと言って営業マンに慣れない5250画面での入力を強いたところで、いたずらに時間ばかりかかるだろうことが容易に想像された。【図2】

既存の RPG で開発されたアプリケーションを活かしつつ、若い世代の社員にも違和感なく受け入れてもらえる、そんな都合のいいツールはないだろうか。このような模索を始めてしばらくした頃、一見相反すると思われるこの 2 つの条件を満たしてくれる Delphi/400 に出合うことができた。

受注入力・在庫照会画面 を同時に GUI 化

初めての GUI 化にあたっては、RPG の開発は自社で、Delphi/400 の開発はミガロ. でという分担で、開発日数を圧縮することを最優先事項とし、受注入力・在庫照会に関連する全てのプログラムを約3カ月で完成させた。でき上がった画面は24×80 の呪縛から解放され、操作性において格段の進歩を遂げることとなった。その後も社内からの要求はとどまることがなく、開発者を1名増員し、受注入力・在庫照会画面にも幾度となく手を加えていった。【図3】

たった1つの照会プログラムが起爆剤となる

在庫照会の GUI 化はさらなる要求に 拍車をかけることとなり、照会系プログ ラム開発の量産へとつながっていった。 【図 4】現在では、1カ月数本のペースで プログラム開発を行うまでになってい

Delphi/400 の開発を 通して学んだこと

たとえ不十分であっても、まず形にして見せること。面倒ではあるが、そうすることによってさまざまな要求が生まれ、アプリケーションも進化していく。社内システムにおいて、この手法はとても効果的だということを体感でき、将来の展望も開けてきた。今後は既存のRPG資産を活かしつつ、DB2/SQLをメイン言語とし、Delphi/400で100%GUI化を目指して開発に取り組んでいきたい。【図5】

M

図1 5250在庫照会



図2-1 5250受注入力



図2-2

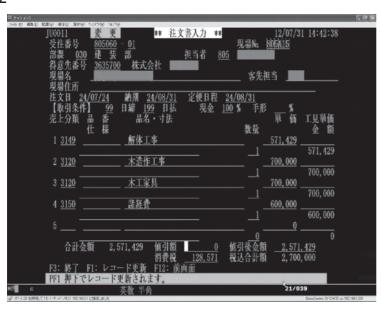


図3-1 Delphi在庫照会



図3-2 Delphi受注入力



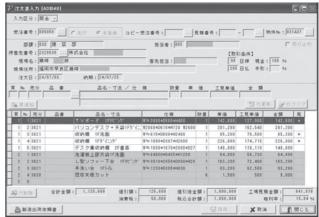
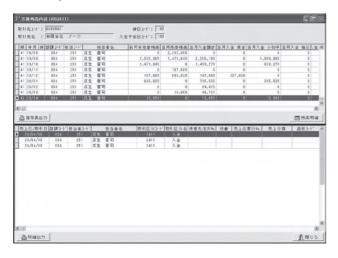


図4-1 Delphi売掛照会



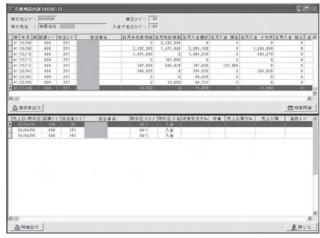
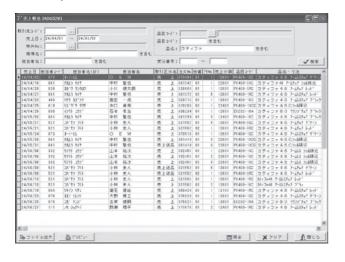


図4-2 Delphi 売上照会



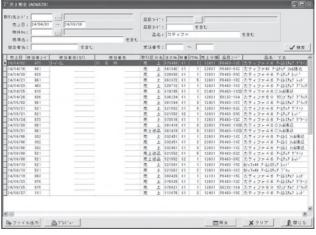
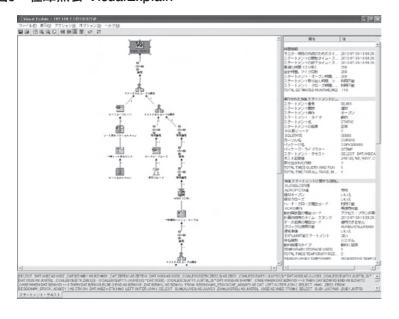


図4-3 Delphi取引先照会





図5 在庫照会 VisualExplain



Delphi/400で確認業務の効率化

為国 順子 様

ベネトンジャパン株式会社 IT システム Div. チーフ



ベネトンジャパン株式会社 http://www.benetton.jp/

1982 年の市場参入以来、多くの日本人に愛用されるベネトンブランド。1987 年には、日本ならではの多様なニーズに応えるべくライセン事業にも着手。世界有数のアパレルブランド「UNITED COLORS OF BENETTON.」や「SISLEY」を擁し、トレンドに敏感なイタリアンカジュアルを全国の店舗にて提供している。

業務課題

複数のシステムを連携して運用している。その連携が正しく行われているかどうかのチェックを、ユーザーが利用している数値項目の一致で確認していた。ただし自動化されておらず、手動による作業であったため時間がかかっていた。また、一致の確認という地味な作業であるため、作業意欲がわきにくい側面もあった。

技術課題

従来は、2つのシステムの情報をExcel に手入力し、計算式による比較結果をExcel に表示させて確認していた。この理由は、利用する資料が複数あり、かつ項目も多数あったのでフレキシブルなExcel が適していると考えたからである。しかし、利用する資料がExcelであったりクエリーの結果であったりするので、統合した画面を作成するためのベースとして Delphi/400 を活用し、

Excel からの数値取り込みを自動化できないかが技術的課題となった。

技術課題の解決策

Delphi/400 の利用により、IBM i 上のデータについては取得できるが、Excel の情報取得のために、取り込み先フォルダをルール化してプログラムでフォルダ名を作成するようにした。【ソース 1】

業務課題解決と効果

数値チェック画面で、数値チェック開始ボタンを押すだけで、Excel と IBM i側のデータを取得して結果を表示するようになったので、作業が1クリックで終了できるようになった。従来は資料を開いて該当箇所を探し、また確認用 Excelの入力位置を出して入力ということをいくつかの項目について行うため、少なくとも15分はかかっていたのだが、数秒で終わるようになり作業効率が大幅に向

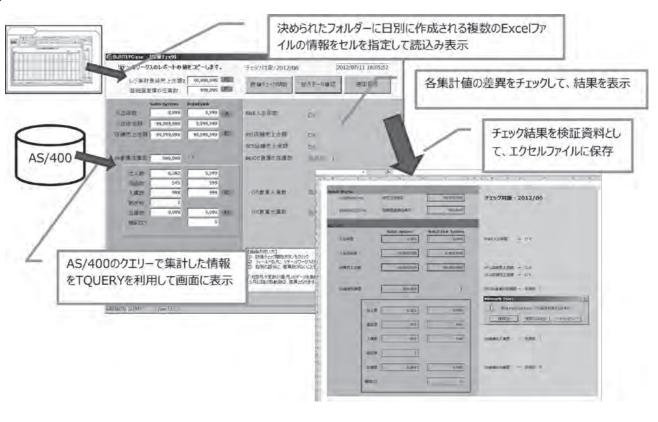
上した。また、精神衛生上もよくなった。 【図1】

M

ソース1

```
Eprocedure TfrmProc019.Button1Click(Sender: TObject);
      dif1, dif2, dif3, dif4, dif5, dif8 : Integer;
    begin
    Today := Date;
190
      DecodeDate(Today, y.m.d);
      sFolder := Copy(Datetostr(today), 1, 4) + Copy(Datetostr(today), 6, 2)
                 + Copy(Datetostr(today), 9,2);
      // 当月の日數を取得
 ū
      EndofMonth := IncMonth(Today, 1) - d;
 ā
      DecodeDate(EndofMonth, y,m,d);
 Ť
      if edit1.text='0' then
200
      begin
      //* レジ葉計表
 W.
      FName := CommonDM.iniDailyCheck1 + '\'
                  + sFolder + '\regishukei.xls';
    // Showmessage (FName) :
 ٠
     Excel := CreateOleObject('Excel.Application'):
      Excel. Visible = True:
4
210
      WorkBook := Excel. WorkBooks. Open (FName);
      WorkSheet := WorkBook.WorkSheets['全店舗計_01'];
      i := d + 12:
      Edit1. Text := WorkSheet. Cells[i, 15]. xalue;
 A
      WorkBook.close;
 ٠
      WorkBook: =unAssigned;
      WorkSheet:=unAssigned;
220
      Excel.Quit;
```

図1



取引先申請システムでの稟議書作成ワークフロー

大崎 貴昭 様

森定興商株式会社 システム運用部 係長



森定興商株式会社 http://www.morisada.jp/

創業は1927(昭和2)年。パイプ問屋としてスタート。現在は、各年のパイプ・鋼材・建材・住宅機器・名庫市の本社のほか、東京・大阪・福岡に拠点がある。2011年に初めての海外現地法人「ベトナムモリサダ」を設立。今後はグローバル展開も視野に入れる。商事部門の他に製造部門・不動産部門がある。

業務課題

取引先との取引開始にあたって、取引 稟議書を担当者から直属上司へ回してい るが、手書きであるため決裁スピードに 欠けている。また、「月次締切まで日に ちが足りない」という理由で承認フロー を無視したり違反する例が見られる。コ ンプライアンスを徹底させたい。

技術課題

当社の Delphi/400 開発者が1名ということもあり、ワークフローのルート変更やユーザー増減の都度、Delphi/400のソースを変更しなくても済むようAS/400のデータベースを使い、ユーザー管理の運用を Delphi/400 開発者以外でもできるようにする。

Delphi/400 では、ユーザー ID とパスワードを入力し【図 1】、関係する取引先の文書を表示させ【図 2】、そこで意見入力をする。【図 3】

文書登録ボタンを押すと、メールの新

規文書作成画面が表示される。承認して もらうユーザーのアドレスをデータベー スから読み取り、宛先に自動表示する。 【図4】

技術課題の解決策

データベースのフィールドには、ID、パスワード、自己アドレス、次承認者アドレスがある。Locate で ID、パスワードのフィールドの内容を読み込む。True であれば申請中 DB から関係する所属のデータを表示する。【ソース 1、ソース 2】

登録ボタンを押すことで申請中 DB の 更新を行う。「お知らせメールを作成し ますか?」のメッセージに対し「はい」 をクリックすると、メーラー(Notes) の起動を確認し【図 5、ソース 3】、起動 していない場合はメッセージを表示する 【図 6、ソース 4】。

Notes が起動していれば、次承認者の アドレスが宛先に表示され、申請者名と 取引先名がパラメーターで渡され本文に 表示される。【図4】

業務課題解決と効果

Delphi/400 による取引先申請システムが完成し運用を開始したところ、ユーザーの中に、ワークフローを厳守しないと取引先との取引が開始できないという意識が芽生えたようである。そして現在、承認のルールは100%厳守されるようになっている。これまで、ややルーズな点もあった取引先の管理であるが、Delphi/400 によるシステム化で改善され、よかったと思っている。

M

図1

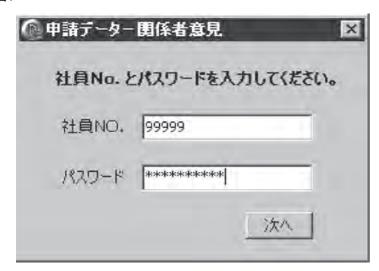


図2



図3

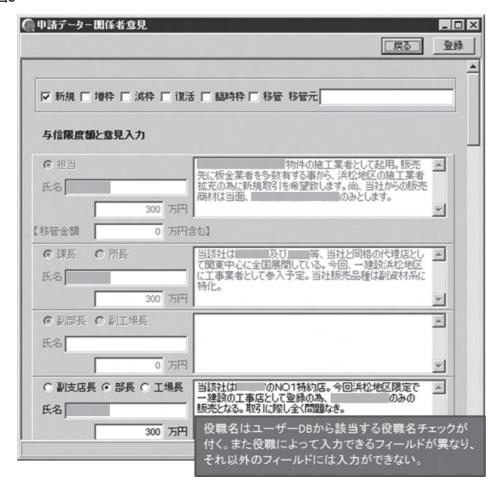
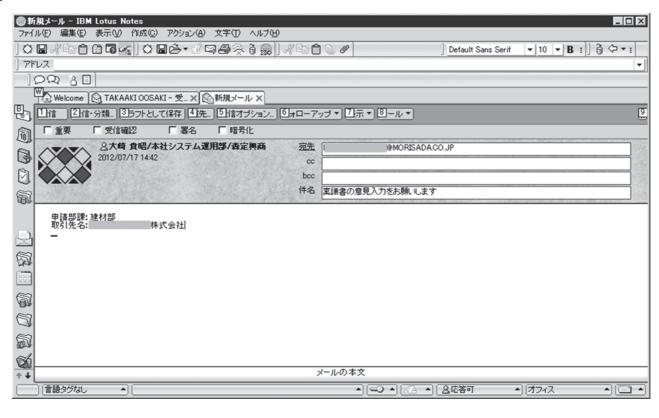
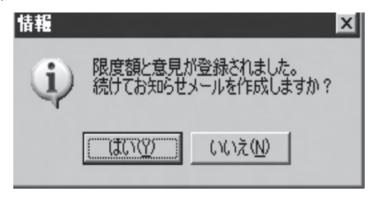


図4



```
ソース1
      *********
           次へをクリック
×部課コードで絞込み
      1/1
           id · passwordで制限
      ********************
      *********
      procedure TFKairan O.Button1Click(Sender: TObject);
      // r : String;
       i : Integer;
      begin
        if E_no.Text = '' then
        begin
          MessageDlg('idが入力されていません',mtError,[mbOK],D);
          E_no.SetFocus;
         Exit;
       end;
if E_pass.Text = " then
          MessageDlg('passが入力されていません',mtError,[mbOK],0);
          E pass . Set Focus:
         Exit;
        end:
        if not((DM.C_id.Locate('BESYID',E_no.Text,[]) = True)
        and (E_pass.Text = DM.C_id.FieldByName('BEPW').AsString))then
        begin
          MessageDlg('idまたはpassが間違っています。',mtError,[mbOK],0);
         E_pass.Text := "
         E_no.SetFocus;
         Exit:
        end;
ソース2
      FKairan 2.inputid := E no.Text;
      DM.C sinsel.Filtered := False;
    // r :=
   StringReplace(DM.C_id.FieldByName('BEBUC').AsString,'**','*',[]);
DM.C_sinsei.Filter := '(KSPRK2 = 1) and (KSBUC1 = ''' +
DM.C_id.FieldByName('BEBUC').AsString + ''';
if DM.C_id.FieldByName('BEBUC2').AsString <> '' then
    DM.C_sinsei.Filter := DM.C_sinsei.Filter + 'or KSBUC1 = ''' +
DM.C_id.FieldByName('BEBUC2').AsString + ''';
// if DM.C_id.FieldByName('BEBUC3').AsString <> '' then
          DM.C sinsei.Filter := DM.C sinsei.Filter + 'or KSBUC1 = ''' +
   DM.C_id.FieldByName('BEBUC3').AsString +
    // if DM.C_id.FieldByName('BEBUC4').AsString <> '' then
        DM.C_sinsel.Filter := DM.C_sinsel.Filter + 'or KSBUC1 = ''' +
   DM.C_id.FieldByName('BEBUC4').AsString + ''';
      DM.C sinsei.Filter := DM.C sinsei.Filter + ')':
   // ShowMessage(DM.C_sinsei.Filter):
      DM.C sinsei.Filtered := True;
    // ShowMessage(IntToStr(DM.C sinsei.RecordCount));
      if DM.C_sinsel.RecordCount = 0 then
        ShowMessage('該当するデータはありません。');
        E no.SetFocus;
                                 入力されたユーザーIDに関係するデータのみを表示する。関係する
       Exit;
      end;
                                 かどうかはAS/400のDBの所属コードを絞り込んで表示するようにして
                                 いる
```



ソース3

```
********
        登録ボタン
*******************
********
procedure TFKairan_2.Bt_tourokuClick(Sender: TObject);
var
   add3 : String;
begin
FKairan_1.Ins;

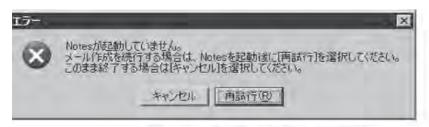
If (DM.C_id.Locate('BESYID',inputid.[]) = True) then

add3 := DM.C_id.FieldByName('BEAD3').AsString;

if (DM.C_id.Locate('BESYID',inputid.[]) = True) and

(DM.C_id.FieldByName('BEAD2').AsString <> '') then
   begin
if MessageDlg('限度額と意見が登録されました。'+ sLineBreak +'続けてお知らせメールを作成しますか?',mtInformation,[mbYes,mbNo],D)=mrYes
then
        FWenu.mail(DM.C_id.FieldByName('BEAD2').AsString,
DM.C_id.FieldByName('BEAD3').AsString,
                    DM.C_sinsei.FieldByName('KSSINK').AsString,
DM.C_sinsei.FieldByName('KSTRI1').AsString);
else if (DM.C_id.Locate('BENAME',tantoid,[]) = True) and (DM.C_id.FieldByName('BEAD1').AsString <> ') then
  begin
        if MessageDlg('限度額と意見が登録されました。
',mtInformation,[mbOK],D)=mrOK then
if MessageDlg('限度額と意見が登録されました。'+ sLineBreak +'続けてお知らせメールを作成しますか?',mtInformation,[mbYes,mbNo],D)=mrYes
then
       FMenu.lastmail(DM.C_id.FieldByName('BEAD1').AsString.
                       DM.O_id.FieldByName('BEAD3').AsString,
11
                    DM.C sinsei.FieldByName('KSSINK').AsString,
                    DM.C_sinsei.FieldByName('KSTRII').AsString);
   end:
   DM.C_sinsel.Active := False;
DM.C_sinsel.Active := True;
   Hide:
   FKairan_I.Show;
end;
```

申請中DBから次承認者のアドレス(BEAD2, BEAD3)、申請者名(KSSINK)、取引先名(KSTRI1)を 渡す



メーラーが起動していない場合のエラー表示

```
ソース4
```

```
********
       Notesに新規メール作成
%90%5C(SJIS) = '申' (文字化けするため)
***********************
*********
procedure TFMenu.mail(toadd,ccadd.bu.office: String);
  HWND, HAPP: THandle;
  strExeName: String;
begin
  Application Initialize;
// メインウィンドウを探す
HWND := FindWindow('NOTES', nil);
                                                                                    メーラーが起動しているかを確認
   while HWND = 0 do
                                                                                     (弊社ではNotesを使用)
   begin
     if MessageDig('Notesが起動していません。'+sLineBreak
+'メール作成を続行する場合は、Notesを起動後に[再試行]を選択してください。'+sLineBreak+ 'このまま終了する場合は[キャンセル]を選択してください。',mtError,[mbCance],mbRetry],0) = mrCancel then
       Exit;
     HWND := FindWindow('NOTES', nil);
   // メインウィンドウがある場合、アブリケーションのハンドルを取得
     HAPP := GetWindow(HWND, GW_OWNER);
     if (IsIconic(HAPP)) then
     // アイコン化されていた場合、元のサイズにする
     OpenIcon(HAPP);
     // 前面に移動
    SetForegroundWindow(HAPP);
strExeName := 'mailto:' +
toadd;//DM.C_id.FieldByName('BEAD2').AsString; //アドレス
{ strExeName := strExeName + '?subject=取引先稟議書'; //件名
// strExeName := strExeName + '&cc=' + ccadd; //cc
strExeName := strExeName + '&body=稟議書の関係者意見の入力をお願
いします。
     strExeName := strExeName + '%Od%Oa' + '';
strExeName := strExeName + '%Od%Oa' + '%90%5C請部課:' +
bu://DM.C_sinsei.FieldByName('KSSINK').AsString;
// strExeName := strExeName + '%Od%Oa' + '%90
DM.C_sinsei.FieldByName('KSSINS').AsString;
strExeName:= strExeName + '%Od%Oa' + '取引先名:' +
Office;//DM.C_sinsei.FieldByName('KSTRI1').AsString;
// strExeName:= strExeName + '%Od%Oa' + '';
// strExeName:= strExeName + '%Od%Oa' + ''※取引先%90%5C請メニュー
                         ※「申」という文字が文字化けするので、S-JISコードを直接指定している
```

Delphi/400でIBM iのストアードプロシージャを利用し、SQL処理を高速化

島根 英行 様

シルフ



シルフ

業務内容はIBMiを中心としたシステム開発。販売管理をメインに、オフコンやPCサーバーからのリプレース提案を得意としている。

業務課題

入出庫のような在庫データの更新処理の際、SQLで一度に更新するデータ量が多いと時間がかかり、現場の作業効率の低下につながる。そこで、パフォーマンス向上のために SQL からストアードプロシージャを利用することにした。これに伴い、複数の人が同時に在庫更新を利用してもデータの整合性を保つための仕組みづくりが必要となった。

技術課題

データの整合性を保つためには、SQLから IBMiのストアードプロシージャを呼び出した際、排他処理とトランザクション制御の実装が必須である。また、排他などのエラーが起きた場合、IBMiのエラー情報を取得し、ユーザーに通知する機能の開発が必要となってきた。

技術課題の解決策

DB2/400 のエラー取得を、条件ハンドラー「DECLER ~」のストアードプロシージャで実施。SQL からこのストアードプロシージャを呼び出し、デッドロックまたはタイムアウトの「40001」というエラーが戻ってきたら、ユーザーに「レコードロックエラー、もう一度実行してください。」というメッセージが表示されるようにした。【ソース1】【ソース2】

業務課題解決と効果

SQL での DB 更新の際、ストアード プロシージャによる更新へ変更したこと により、 $7000 \sim 8000$ 件の一括データ更 新の時間が 1/3 程度まで短縮され、ユー ザーの作業効率が向上した。

また、ユーザーへエラー表示もできる ようになり、更新エラー時は再処理を行 うよう促すことが可能になった。

ストアードプロシージャの利用によ

り、以下のようなメリットを得ることができた。

- ①ネットワーク負荷を軽減できる。
- ②構文解析や内部中間コードへの変換を 行うため、処理時間が軽減される。
- ③ Delphi 言語と SQL とで分業が可能 になる。

M

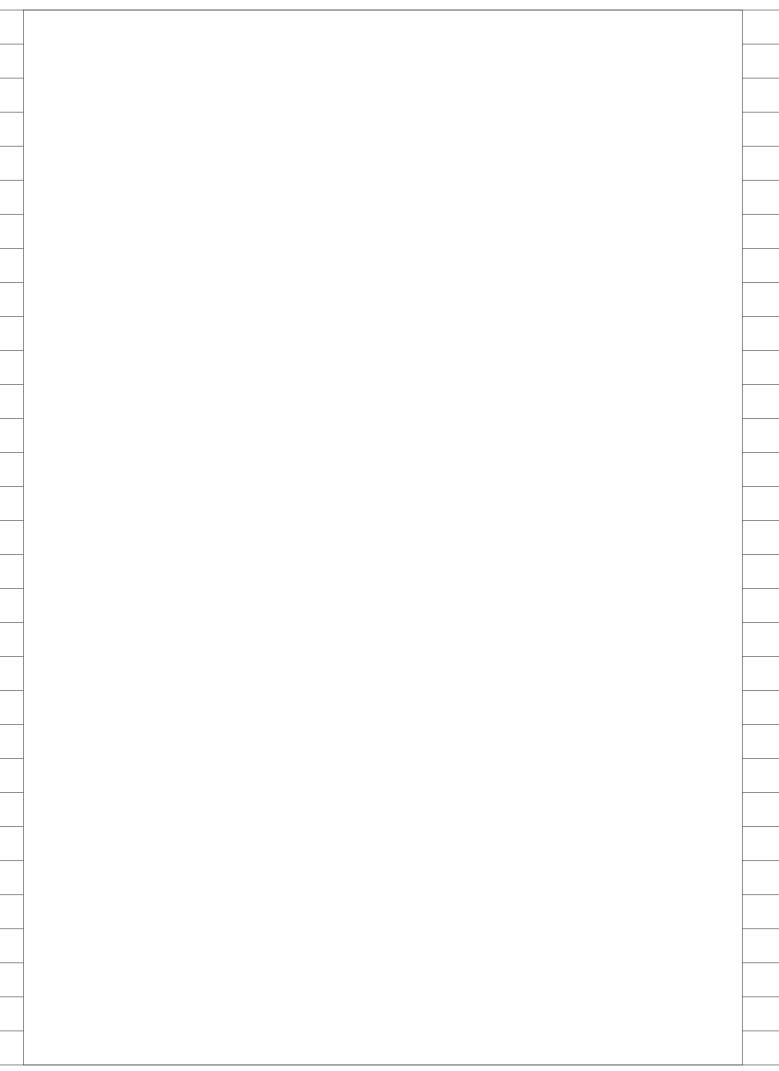
ソース1

```
## StoredProcl.StoredProcName := 'CHKZAIKO';

| StoredProcl.Params.Clear;
| StoredProcl.ParamByName('CHKZADO1').AsAnsiString := ZADD1;
| StoredProcl.ExecProc;
| Chkzaiko := StoredProcl.Params[1].AsInteger; //在庫數
| LoutSQLSTATE := StoredProcl.Params[2].AsString; //SQLCQDE
| //デッドロックまたはタイムアウト
| if chkSQLSTATE=40001 then begin
| DB2.Rollback;
| MessageBox(self.Handle, 'レコードロックエラー、もう一度実行してください。
| **Msg', MB_OK);
| Exit;
| end;
| end;
```

ソース2

```
//ストアード・プロシージャー
    CREATE PROCEDURE CHKZAIKO
       IN CHKJANOO1 CHAR(13),
      OUT ZAIKOI NUMERIC(7,0),
      OUT SSTATE CHAR(5)
    RESULT SETS 1
    LANGUAGE SQL
MODIFIES SQL DATA
10
    BEGIN ATOMIC
      DECLARE ZAIKO NUMERIC(7,0);
DECLARE SQLSTATE CHAR(5) DEFAULT '00000';
      DECLARE C1 CURSOR FOR
        SELECT kazu
        FROM zaiko
        WHERE
        JAN001=CHKJAN001;
    --条件ハンドラー
DECLARE UNDO HANDLER FOR SQLEXCEPTION /*UNDO ロールバック */
20
     BEGIN
       SET SSTATE =SQLSTATE; /*SQLCODE*/
     END;
      OPEN C1;
      FETCH FROM C1 INTO ZAIKO;
      CLOSE C1;
      SET ZAIKO1=ZAIKO;
29
    END
```





創刊 5 周年記念 ミガロ.SE 座談会 お客様と共に歩む お客様への熱い思い ミガロ.SE座談会

お客様と共に歩むお客様への熱い思い

ミガロ.のエンジニアは、お客様の業務に役立つことを業務の真髄と考え、システムの開発・保守と製品のサポートに日々、取り組んでいます。開発・保守や製品の拡張・メンテナンスを通して個々の SE が得た Delphi/400 や JC/400、Business4Mobile の凄さ、そしてお問い合わせやサポートを通じてお近づきを得たお客様への思いを、ミガロ.の SE 5 名が語り合います。

出席

松尾悦郎

システム事業部 部長

田村洋一郎

システム事業部 システム 1 課 課長

尾崎浩司

システム事業部 プロジェクト推進室 課長

吉原泰介

RAD 事業部 技術支援課 課長

畑中 侑

システム事業部 システム 2課 主任

入社後の経歴と 仕事内容

— 最初に、入社後の経歴と現在の仕事内容をご紹介ください。

吉原 私は、入社してから2年ほどシステム事業部でDelphi/400を使ったお客様システムの開発を担当しました。その後、RAD事業部へ異動して製品担当となり、現在は、Delphi/400やJC/400などミガロ.製品をご購入いただいたお客様のサポートを担当しています。また、そのかたわら、製品の検証やイベント/セミナーの講師なども務めています。

尾崎 私は 1999 年の入社で、ちょうど丸 13 年経ちました。入社以来、Delphi/400 一筋で、主にお客様からの受託開発を担

当しています。Delphi の最初のリリー スが 1995 年ですので、Delphi の成長と ともに技術者としての経歴を重ねてきた 形です。販売管理や見積管理システムを 数多く手がけ、本日の座談会メンバーの 中では最も多くの開発プロジェクトを経 験しているかと思います。Delphi/400 を使った Web システムの開発も、ミガ ロ.で最初に担当させていただきました。 畑中 私は入社7年目で、Delphi/400 を使ったお客様システムの開発や保守 を担当しています。大小さまざまなプ ロジェクトに入りますが、大きなプロ ジェクトではサブ・リーダー、小規模 な案件ではリーダーを務めることが多 くなりました。現在は、私の入社前か ら Delphi/400 をお使いのお客様の保守

を担当していますが、お客様がこうあり たいと思われたシステムを実現してきた Delphi/400 は素晴らしいなと感じてい るこの頃です。

田村 私は畑中と同期で、入社7年目となります。お客様の基幹システムをDelphi/400やJC/400を使って開発する仕事を担当しています。役割としては、プロジェクトマネージャーやプロジェクトリーダーとなります。最近ではプログラミングをすることが、ほとんどなくなりました。

松尾 私も畑中・田村と並んで入社7年 目ですが、転職組ですので、そのぶん年 長です。前職は製造系企業で自社システ ムのSEをやっていました。その時の言 語がRPGで、ミガロ.入社後もRPG関 連の開発を数多く担当することになりました。当社の製品では、Delphi/400よりもJC/400のほうを多く手がけています。現在はプロジェクト・マネージャーとしてプロジェクト全体の責任を負っていますが、人手の足りなくなると開発側に回ることも稀にあります。それと、システム事業部全体の管理も担当しています。

Delphi/400 の魅力 その奥深さ

一 皆さんは、Delphi/400をお客様へ 提供する側ですが、同時に Delphi/400 のユーザーでもあります。そうしたへ ビーユーザーから見て、さらに製品の裏 側もよく知っているお立場からして、 Delphi/400の魅力は何だと感じていま すか。

吉原 私が思うに、Delphi/400の一番いいところは、業務システムのWeb/GUI画面を簡単に作れるのと同時に、RPGで作成したプログラムを生かして業務ロジックを開発できる点ですね。つまり、画面と業務ロジックとに分けて、分業で開発できるのが大きな魅力です。実際、RPGを長く使ってこられたお客様がDelphi/400を導入すると、そうした分業や使い分けをごく自然に実践されています。

松尾 ミガロ.の開発が、まさにその分 業体制ですね。それが効率的な開発につ ながっています。

尾崎 それと、開発ツールとして軸がぶれないのも大きな魅力です。その点はRPGと一緒ですね。RPGはILEへ進化しても、あるいはWebやSOAに対応しても、まったく軸がぶれません。それがRPGの特質であり魅力だと思いますが、Delphi/400もGUIからWebへ、モバイル対応へと進化し続けていますが、ツールとしての軸は当初のままです。VBが.NETになった時のような大きな方針変更はありません。その点は開発者にとって魅力で、信頼のおける部分ですね。——Delphi/400の進化で印象に残っているものはありますか。

尾崎 過去の機能拡張で驚いたのは、Webに対応した時ですね。衝撃を受けました。当時(2003年)、Webシステムの開発というと Java の習得が必須と言われていて、ちょっと敬遠していたん

です。Web よりも GUI のほうが優れているとも思っていました。そうした中で Delphi/400 の Web 対応 機能 が出 て、今までの作り方をそのまま踏襲して、いとも簡単に Web システムを開発することができた。「また一歩進化したな」と思い、感激した覚えがあります。

畑中 私は、IBM i との連携が素晴らしいと感じています。画面は Delphi/400で簡単に作れます。もちろん、アイデアを盛り込んだ、凝った画面も可能です。それでいて IBM i とのやり取りは完璧です。

— Delphi/400 は誰かに教えてもらっ たのですか。

畑中 吉原さんが先生でした。おかげさまで、こんなに立派に成長しました(笑)。 吉原 Delphi/400の扱い方やプログラミングの習得は、それほど苦労はしなかったですね。むしろ大変だったのは、お客様の話す内容を理解して要件をまとめていくほう。これは苦労していましたね。

田村 私の Delphi/400 の師匠も吉原さんで、畑中を含め同期の 4 人全員が吉原さんに教えを乞いました。何かあると 4 人それぞれが入れ替わりで吉原さんに質問するので、吉原さんのほうは自分の開発の仕事がなかなか進まなかったと思います。

私は習得が遅かったので、他の3人に比べ特に手間がかかったと思います。ただ、私が恵まれていたのは、プロジェクトでベテランの開発者が周囲にたくさんいて、手取り足取り、いろいろなことを教えてもらえたことです。その中で、当時、「Delphi/400 といっても開発言語の1つだからね」と言われた瞬間に、それまで難しいと感じていた Delphi/400 に対する苦手意識がすっと消える経験をしました。

今は、お客様とお話しながらお客様向けシステムの提案を行っていますが、Delphi/400で実現できることは幅広く、複雑なシステムにも対応できる点が凄いと感じています。そこが Delphi/400 の強みであり、魅力です。お客様に自信をもってシステム化をご提案できるツールですね。



パートナーとしての関係を結びたいですね。お客様と開発業者という関係を超えた、

Delphi/400 だから ご要望に対応できた

畑中 本当にそう思います。同感ですね。 私のほうは、Java と Excel で作られていた営業支援システムを、Delphi/400で再構築する仕事を担当したことがありました。基本的な機能を変えずにビジュアル面を大幅に変更し、使い勝手とパフォーマンスを改善したのですが、お客様にとても喜んでいただき、高い評価をいただきました。これなど、Delphi/400だからこそ、お客様のご要望をうまく実現できたと思うんですね。

田村 私も、システム化が難しい業務を Delphi/400 で解決できた経験があります。お客様の業務目的を考えるとパッケージでの実現は難しく、システム的にどう実現するか、あれこれ考えた挙句、 Delphi/400 を採用し、完成に漕ぎ着けました。実現性の高い Delphi/400 を採用してよかったな、と今振り返っても思いますね。

松尾 RPG技術者の方々は、オープン系の言語やツールが導入されると自分たちの居場所がなくなるような感覚を持た

れていると思うんです。それは、オープ ン系の言語/ツールと RPG が交わらな いからですが、Delphi/400 はオープン 系ツールでありながらそういう懸念が全 く不要です。先ほど吉原さんが触れたよ うに、分業が可能だからです。なので、 RPG技術者にとって Delphi/400 は、 オープン系と RPG を橋渡しする魅力的 なツールになるではないかと思います。 尾崎 RPG は、ロジックが書けるよう になるまで習得に少し時間がかかります が、Delphi/400 は簡単なことならすぐ に開発できます。初めてのお客様には、 「まず勇気を持って Delphi/400 に触っ てみてください」と申し上げたいですね。 そのうちに、Delphi/400の奥深いとこ ろも見えてくるかと思います。

Business4Mobile & JC/400 の魅力

松尾 Delphi/400 だけでなく、JC/400 の魅力も言わせてください(笑)。

―― 失礼しました (笑)。どうぞ。 松尾 JC/400 は、RPG 技術者の視点で



お客様をサポ する時 は、

ノロジェクトの部外メンバー」という気持ちです。

いうと、RPG で Web システムを作れる 気持ちにさせてくれるツールです。わず か4ステップだけで、非常に簡単に Web システムを開発できるのですが、 それだけでなく、画面を細かく作り込め るなど、凝ったことも可能です。RPG だけでWebシステムを構築できる点は、 他のツールにない魅力じゃないかと思い ます。

吉原 私は、Business4Mobile に触れ たいですね。昨年からスマートデバイス の業務活用が大きな話題になっています が、実際に業務システムで使っていると ころは、まだそれほど多くないと思いま す。Business4Mobile は開発ツールで はなく、参照ツールの位置づけです。デ バイスを選ばず、どのスマートデバイス にも対応します。そして、導入してすぐ に使えます。そこが魅力ですね。スマー トデバイスで社内 DB を活用する第一弾 のツールになるかと思います。

お問い合わせと サポートに対する姿勢

- お客様からのお問い合わせ、サ ポート要請に対しては、どのように対応 していますか。

松尾 まず、お問い合わせをいただいた ら、すべてにご対応するのが基本です。 中には、IBM iの OS やシステムに関す るような、私どものテリトリーとは異な るご質問もあるのですが、極力対応する ようにしています。それは、お問い合わ せや、サポート要請をしてこられるのは、 お客様が困っておられるからですね。何 とかして差し上げようという気持ちにな ります。

尾崎 開発したシステムをお客様にをお 引渡しした後は、サポート要請はそれほ ど多くありません。どちらかと言うと、 今後こんなことをしたいがどう思うか、 という内容が多いですね。

田村 私も同じですね。お客様から「社 内で業務改善をしようと検討している が、何かいいシステム化案はないか?」 といったお問い合わせを頂くことがあり ます。お客様と一緒に業務改善の提案を させて頂く際、自分の存在意義を実感で き、ありがたいと感じます。

畑中 納品したシステムの仕様を問い合 わせいただくことがあります。マニュア



Delphi/400 のファンを増やしていきたいです が本当に好きです。

ルを開くよりミガロ.に聞いたほうが早 くて正確だと思っておられるからでしょ うが、歓迎ですね。

尾崎 お客様のほうも、システムを自分 で動かしてから見えてくるものがあるん ですね。納品した直後には、仕様や操作 についてのご質問がよくあります。そう した時に、こうなっているとさらによ かった、という感想があった場合は、仕 様変更や改修にならない限り、修正しま す。Delphi/400 はそうした変更にも柔 軟に対応できるのがいいですね。

吉原 私の方は製品サポートなので、大 半は顔の見えないお客様からのお問い合 わせです。月に100件程度、年にする と約 1000 件ほどのお問い合わせを受け ます。当社のサポートは、電話でもメー ルでもお受けするのが特徴ですが、電話 が2~3割、メールが7~8割といっ た割合です。中には、開発の途中で行き 詰まった時の画面を逐一キャプチャして 送ってこられるお客様もいます。そうし た時は、こちらも逐一対応してご説明し ます。そうして、そのお客様から「リリー スできました」という報告をいただくと、 何ともうれしい気持ちになりますね。も

う「プロジェクトの部外メンバー」とい う気持ちでサポートをしています。

松尾 かなり昔のことになりますが、イ ンシデント制から無制限にサポートのポ リシーを変更しましたよね。

吉原 ええ。その心は、製品のサポート ではなく、ミガロ.の製品を活用してい ただくためのサポートなのだから、困っ ておられることがあればいくらでもお手 伝いしようということですね。

技術者としての夢 かくありたい将来像

―― 最後に、技術者としての夢をお話 ください。

畑中 一言で言えば、頼られる技術者で す。システムの仕様から今後の構想まで 何でも相談していただける、また、その 期待に応えられるエンジニアですね。そ のためには技術を磨き、知識を蓄え、人 としての魅力を増していかねばと思って います。

田村 私は、お客様の企業価値を高めら れる SE です。コンサルタント的な要素

Delphi/400 は、

システム化をご提案できるツールですね。 お客様に自信をもって

田村洋

郎

を提案できる SE でありたいと願ってい ます。私は、ミガロ.の価値は、お客様 の立場に立ってシステム開発するという 強い気持ちにあると思っていますので、 それを体現するエンジニアが自分の将来 像ですね。

松尾 私のほうは、お客様と開発業者と いう関係を超えた、パートナーとしての 関係を結べる技術者でありたいと思いま す。信頼を置いていただけるパートナー として仕事をしていきたいですね。

吉原 私は常に新しい技術にチャレンジ し、それを吸収するとともに情報発信も して、お客様をリードするエンジニアで ありたいと念じています。お客様が安心 してご利用になれる環境を提供するのが 私たちの価値であり、誇りと思っていま す。

尾崎 私は本当に Delphi/400 が好きな ので、Delphi/400のファンを増やした いですね。私は、Delphi/400で一番の 技術者になろうと思ってこれまで努力し てきましたが、それはこれからも変わり ません。今後は、Delphi/400の仲間を 増やすべくセミナーでお話したり、この テクニカルレポートに投稿していきたい と考えています。

M

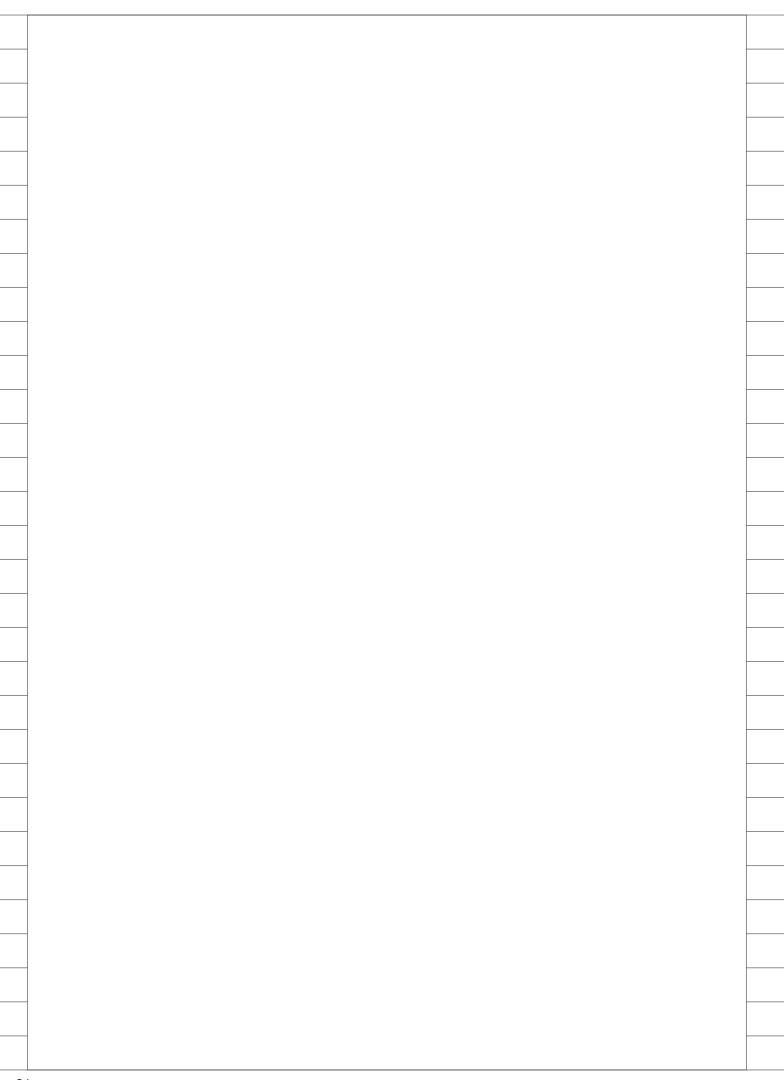


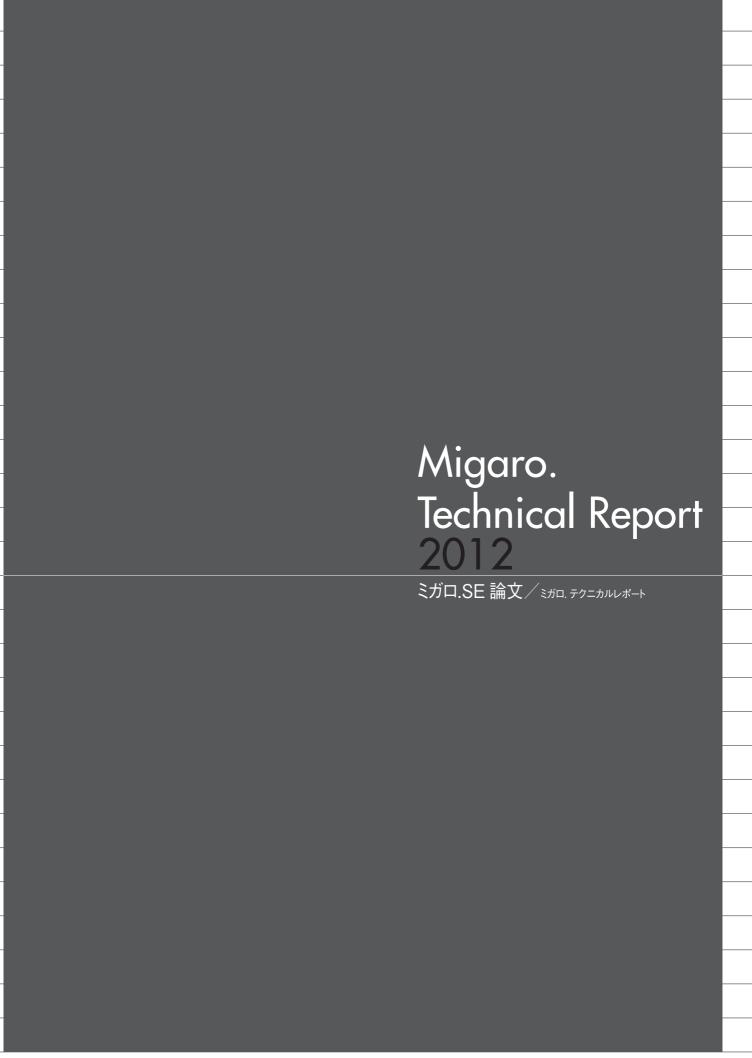
頼られる技術者」を目指します。

技術を磨き、

知識を蓄え、人としての魅力を増して

畑 中 侑





中嶋 祥子

株式会社ミガロ.

RAD事業部 技術支援課 顧客サポート

InstallAwareを使った Delphi/400運用環境の構築

インストーラ作成ツール「InstallAware」。ファイルの配布から、Delphi/400 のインストーラを呼び出す方法までを紹介する。

- ●はじめに
- ■InstallAwareの基本操作
- ●dbExpressでのポイント
- ●BDEでのポイント
- ●Delphi/400インストーラの呼び出し
- ●補足および注意点
- ●まとめ



略歴 1968年02月23日生 1990年奈良女子大学家政学部卒 2002年株式会社ミガロ. 入社 2002年11月RAD事業部配属

現在の仕事内容

お客様からの Delphi/400 に関する 技術的な質問や問い合わせに対応し ている。また、メールマガジン [Migaro News] やホームページの Tips など、開発に役立つ情報も担 当している。

1.はじめに

Delphi/400 を使用したクライアント サーバ型アプリケーションを配布する場 合、運用端末には、アプリケーションファ イル (exe など) や使用している BDE、 dbExpress といったデータベース接続 に必要なファイル等を配布する。

本稿では今回、それらファイルの配布 と Delphi/400 運用版のインストールを 行う方法を、Delphi/400 Version2007 以降のバージョンに付属しているインス トーラ作成ツール「InstallAware」を 用いて実現する方法を紹介する。

2.InstallAwareの 基本操作

まずは、InstallAware の一般的な操 作方法を簡単に説明する。

InstallAware を起動した状態が図1である。初回は、プロジェクトウィザードが起動する。起動しない時には、メニューの[ファイル|新規|デフォルト

のプロジェクト] から表示することができる。【図 1】

「プロジェクト名」と保存先の「Project フォルダ」を指定し、[OK] ボタンを押すと、作成画面に移動する。

画面構成は左にツリーメニューがあり、右がそれに対応した設定画面となっている。それでは次項から、必要最小限の設定項目について、ツリーメニューを上から順に説明していこう。

2-1. アプリケーション情報

アプリケーション情報では、インストーラの基本情報を設定する。図2は「プロジェクトのプロパティ」画面である。 【図2】

プロジェクトウィザードで指定したプロジェクト名が「製品名」に表示される。「ターゲットフォルダ」はデフォルトで \$PROGRAMFILES \$¥\$TITLE \$となっている。このターゲットフォルダは、後述するがファイルの配布先指定時のフォルダになる。

具体的には、InstallAware は配布先のフォルダを指定する場合には、\$と\$で囲まれた予約語で行う。
\$PROGRAMFILES \$も\$TITLE \$も予約語であり、\$PROGRAMFILES \$は C:\(\fomage Program \) Files フォルダを、
\$TITLE \$は製品名を指す。つまり、製品名が SampleInstaller とすると、ターゲットフォルダは C:\(\fomage Program \) Files\(\fomage Sample Program \) Sample Installer となる。なお、もちろん予約語を使用せずに、値を C:\(\fomage Migaro のように固定値とすることもできる。

その他、フォルダを指定するいくつかの予約語を図3に示す。【図3】

2-2. セットアップ・アーキテクチャ

セットアップ・アーキテクチャでは、 アプリケーション等の配布するファイル を指定する。図 4 は「ファイル」画面で ある。【図 4】

画面は4つに分かれており、ファイルの指定は画面上部の①②から行う。また配布先は、画面下部の③④になる。



図2



予約語	フォルダ (XP)	フォルダ(VISTA/Windows7)
\$WINSYSDIR\$	C.¥WINDOWS¥System32¥	
\$COMMONFILES\$	C:¥Program Files¥Common Files¥	
\$DESKTOPDIR\$	C:¥Documents and Settings¥(ユーザー名)¥デスクトップ¥	C:¥Users¥(ユーザー名)¥Desktop¥
\$MYDOCUMENTS\$	C:¥Documents and Settings¥(ユーザー名)¥My Documents¥	C:¥Users¥(ユーザー名)¥Documents¥

- ①端末のフォルダー覧が表示される。
- ②①で選択したフォルダの内容が表示される。
- ③配布先のフォルダが、\$と\$で囲まれた 予約語で記載されている。
- ④③で選択したフォルダの内容が表示される。ただし、これは②と異なり、配布するものを表すので、設定を行っていない現時点では何もない状態となっている。

②でファイルを選び、③の配布先フォルダを選択した状態で[ファイルの追加]ボタンを押すと、④に追加される。【図5】

なお、配布先フォルダにある予約語の \$TARGETDIR \$は、前述の「2-1. アプリケーション情報」で指定したターゲットフォルダである。インストール先を変更したい場合には、前述の「プロジェクトのプロパティ」画面(図 2)に戻って変更を行う。

2-3. ユーザー・インターフェース

ユーザー・インターフェースの「ダイアログ」では、インストーラ実行時に表示されるダイアログを選択する。設定画面で各ダイアログ画面を選択すると、横にプレビューされるので、確認しながら選ぶことができる。

例えば、licensecheck は、図のように License Agreement (使用許諾) の画面が用意されている。他にもREADME、進行状態を表すプログレスバー画面などがあるので、必要に応じて選択する。不要であればチェックを外せばよい。【図 6】

また、使用許諾や ReadMe で表示する内容については、ツリーメニューの「EULIA 及び ReadMe」で指定できる。ここでは必要に応じて設定する。

なお、ダイアログは英語となっているが、インストールには影響しないので安心していただきたい。もし日本語化したい場合には、ローカライゼーションを行う必要がある。方法については、Delphi 開発元であるエンバカデロ・テクノロジーズのサイトに方法が記載されているので、そちらを参照していただきたい。

【InstallAware を使って配布モジュール を作成する - インストーラのローカライズ】

http://edn.embarcadero.com/jp/article/34383

2-4. 配置

配置の「ビルドの設定」では、まず、 作成されるインストーラのタイプを指定 する。タイプには、圧縮されていないも のと圧縮されたものがある。【図7】

●圧縮されていないディレクトリ

どのように異なるかを見るために、まずはデフォルトの「圧縮されていないディレクトリ」を見てみよう。

この状態で、メニューの [ファイル | 保存] から一旦プロジェクトを保存し、フォルダ構成を確認してみると、図8のようになっている。なお、これはインストーラのタイプにかかわらず同じである。【図8】

ではここから、「圧縮されていないディレクトリ」タイプでビルドを行っていく。 設定内容からインストーラの作成を行う には、メニューより [プロジェクト | ビルド] を選択する。ビルド中は経過が逐次表示される。【図 9】

正常に終了すれば自動で閉じられる。 エラーがある場合にはメッセージを確認 し、修正を行う。

では、ビルド実行後に再びプロジェクトフォルダを確認しよう。すると、図10のように、Release¥Uncompressedフォルダの下にインストーラが作成されている。このexeがインストーラになる。

なお、exe 以外にもファイルがあるが、これらを参照しながらインストールが行われる。 そのため、CD-ROM 等にインストーラを作成する際には、Uncompressed フォルダ以下の全ファイルが必要になる点に注意しよう。【図10】

●圧縮された Single-Installing EXE

次に「圧縮された Single-Installing EXE」タイプに変更し、ビルドを行ってみる。図 11 のように、Release フォルダの下には、Uncompressed フォルダとは異なり、別の Single フォルダが作成されて 1 つの exe だけが存在している。【図 11】

つまり、圧縮されて、すべてのファイ

ルがこの exe に含まれている状態である。図 10 の「圧縮されていないディレクトリ」タイプと異なり、複数のファイルやフォルダが存在していない。

このため、フォルダで管理する必要がなく、1つのファイルで管理すればよいので、「圧縮された Single-Installing EXE」タイプのほうが便利ではある。ただし、ビルド時に圧縮が行われ、実行時には解凍されるため、配布するファイルのサイズ等によっては時間がかかることもある。

2-5. インストーラの実行

インストーラを実行してみよう。すると、選択したダイアログが順に表示され、ファイルが配布される。【図 12】

3.dbExpressでの ポイント

dbExpress を使用したアプリケーションを配布する場合、必要なファイルは、 Delphi/400 がインストールされた開発環境に存在する。また、必要なファイルは Delphi/400 のバージョンにより異なる。

● VXE、V2010、V2009

VersionXE、Version2010、Version2009 では、次の3つになる。

· midas.dll

配布元・配布先ともに C:\WINDOWS\ System32 である。InstallAware での配布 先指定は、予約語\$WINSYSDIR \$となる。

· dbxadapter.dll

配布先は C\text{\text{YWindows\text{\text{\text{System32}}}} で、InstallAware では予約語\text{\text{\text{WINSYSDIR}}\$ を指定する。また、配布元は Delphi/400 のバージョンにより異なり、以下のようになる。

(VXE)

C:\Program Files\Embarcadero\RAD Studio\R0\Studio\R0\Studio\R0\In in

(V2010)

C:\footnote{\text{Y}Program Files}\footnote{\text{E}mbarcadero}\footnote{\text{R}AD} Studio\footnote{\text{7}.0}\footnote{\text{bin}}

(V2009)

C:\Program Files\CodeGear\RAD

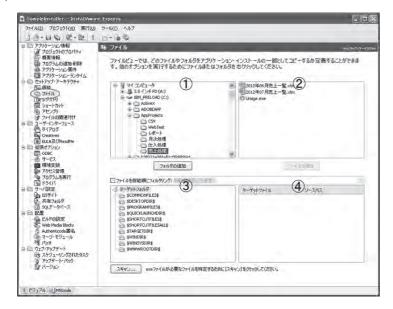
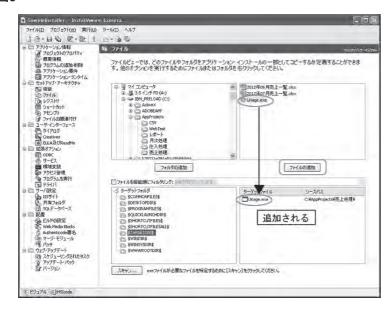
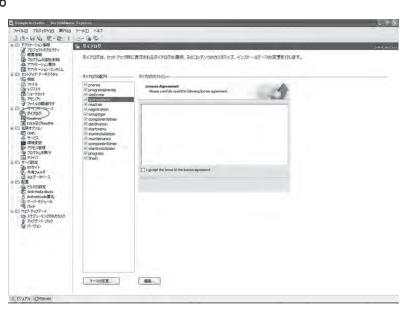


図5





Studio¥6.0¥bin

· dbxdrivers.ini

配布先は EXE と同じフォルダで、InstallAware では予約語\$TARGETDIR \$ を指定する。また、配布元は Delphi/400 のバージョンにより異なり、以下のさらに各フォルダに存在する。

C:\Documents and Settings\All Users\Documents\RAD Studio\documents\RAD S

(VXE)

∼¥dbExpress ¥8.0

(V2010)

∼¥dbExpress ¥7.0

(V2009)

~\dbExpress

V2007

一方、Version2007 の場合、2 つのファイルを配布する。1 つは Version2009 以降と同様に、Midas.dll である。もう1つは dbxadapter30.dll である。

- · Midas.dll
- · dbxadapter30.dll

(V2007)

C:\foramFiles\foramFodeGear\foranteRAD Studio\foranteS0\forante

前述の「2-2 セットアップ・アーキテクチャ」で、これらのファイルの配布を指定する。Midas.dll を指定した場合、図 13 のようになる。【図 13】

4.BDEでのポイント

実は BDE は dbExpress と異なり、ファイルの配布ではなく、マージモジュールを使用して実現している。BDE は多くのファイルから構成されており、マージモジュールを組み込むだけで、各ファイルや配布先の指定も不要でBDE 環境が作成される。

詳細は、以下のサイトを参照してほしいが、マージモジュールは、Delphi 開発元のエンバカデロ・テクノロジーズのサイトよりダウンロードして入手する必要がある。また、ダウンロードする際には、Delphi の使用許諾を取得したユー

ザー名(もしくは email アドレス)と パスワードも必要になる。

【ミガロ, HP

「Delphi/400 Version2007 以降の BDE マージモジュールの追加方法」】

http://www.migaro.co.jp/contents/products/delphi400/faq/tec.html#q47

【登録ユーザー向けダウンロードページ - BDE Merge Module for RAD Studio 2007-XE2】

http://cc.embarcadero.com/myreg

マージモジュールの使用方法は、まず、 ダウンロードしたマージモジュール BDE_PRO.Msm を、適当なフォルダに コ ピ ー す る (C:\Program Files\ Common Files\Merge Modules など)。

その後、InstallAware でマージモジュールを指定する。これは、[配置 | マージモジュール] のステップで、上記のフォルダにコピーした BDE_PRO. Msm を指定する。【図 14】

注意するポイントは、マージモジュールを含めた場合には、ビルドの種類で「圧縮されていないディレクトリ」を選択することである。圧縮されるタイプでは、生成されたインストーラからBDEが正しくインストールされないケースがある。

5.Delphi/400インストーラの呼び出し

InstallAware には、インストールの 前や後などのタイミングで、指定したプ ログラムを呼び出す機能が存在してい る。この機能を利用して、さらに Delphi/400のインストーラ (Setup. exe) を呼び出してみよう。

Delphi/400 のインストーラは CD-ROM にあるが、InstallAwareから CD-ROM を参照する場合、予約語を追加して使用できるようにする必要がある。この予約語により、端末ごとにドライブが異なっても参照可能になる。

CD-ROM を参照する設定を行うには 最初に、前述の「2-2. セットアップ・アー キテクチャ」の「ファイル」のステップ を用いる。「ターゲットフォルダ」上で 右クリックし、「システムフォルダの追加」を選択する。表示されたダイアログ上で「CD-ROMパス」を選択して登録すると、\$MYCDPATH \$が追加される。(ここではターゲットファイルを指定する必要はない)。【図 15】

次に、Setup.exe の呼び出しを指定する。ツリーメニューの[拡張オプション |プログラムを実行]を選択する。実行プログラムの一覧が表示されるので、そこで右クリックし「新規」を選択すると、ダイアログが表示される。【図 16】

このダイアログでは、次の項目を指定する。

・「プログラムファイル」

配布したファイルを指定する場合は [ブラウズ] ボタンから選択するが、今 回のように、CD-ROM の Setup.exe を 指定する場合には、\$MYCDPATH \$\ SETUP.exe と直接入力する。

なお、変数\$MYCDPATH \$を有効に するには、上記の[セットアップ・アー キテクチャ | ファイル]のステップで追 加した\$MYCDPATH \$のフォルダが存 在していなければならない。登録しない まま、ここで直接、実行プログラム名で 指定しても動作しない。

・ 「スケジュールの実行」

今回は「インストール後」とする。なお、BDE の場合は、Delphi/400より先にBDE がインストールされていなければならないため、必ず「インストール後」とする。

・「プログラムが終了するまでお待ちく ださい」のチェックボックス

チェックを入れることで、呼び出した プログラムが終了するまで、インストー ラが待機する。

以上を設定した画面が図 17 である。 【図 17】

ビルドして作成されたインストーラを 実行すると、指定した dbExpress や BDE のファイルの配布後、さらに Delphi/400 のセットアップ画面が起動 する。そこで、通常どおり、画面の指示 に従って進めていくと、Delphi/400 が インストールされる。

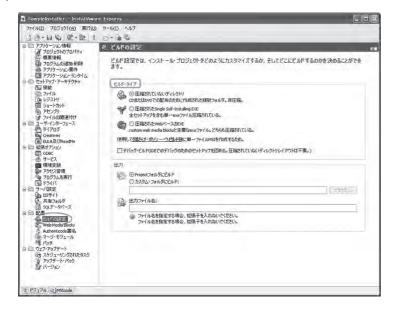
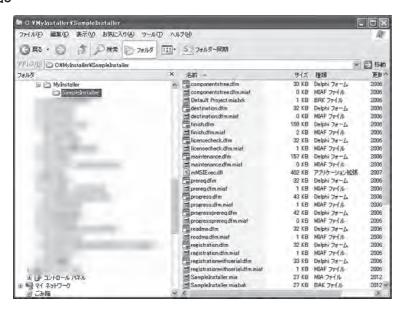
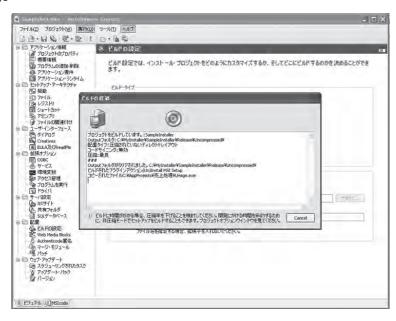


図8





●サイレントインストール

ここで、Delphi/400 のインストーラ 画面を表示せずに、Delphi/400 のイン ストールを実現する方法を紹介しよう。

Delphi/400 運用版は、「サイレントインストール」を行うことができる。サイレントインストールとは、応答ファイルを使用することにより、インストール中にインストーラ画面が表示されず、入力や応答が不要になるインストール手法である。

なお、Delphi/400インストーラ用の 応答ファイル (iss ファイル) は、製品 保守のサービスから入手することができ る。注意点としては、issファイルは Delphi/400のバージョンごとに用意さ れているので、使用バージョンに適応し たものを使う必要がある。

通常サイレントインストールは、Windows のコマンドプロンプトより、以下のコマンドで実行する。このコマンドにより、Dドライブにある Setup.exeが指定された応答ファイル C:\INSTALL\setuppcE.iss を参照して実行され、その間インストール画面は一切表示されない。

D:\$SETUP.EXE -S -F1C:\$INSTALL\$ setuppcE.iss

InstallAware では、ツリーメニューの[セットアップ・アーキテクチャ | ファイル] のステップで、応答ファイル setuppcE.iss の配布を指定する。図 18では、配布先 \$TARGETDIR \$ に setuppcE.iss を指定している。【図 18】

次に [拡張オプション|プログラムを 実行] のステップで、[プログラムファ イル」はパラメータを付けず \$MYCDPATH \$\frac{\text{\$\text{\$YSETUP.exe}}}{\text{\$c}} のままと し、[パラメータ] でパラメータの-S -F1C:[\$\text{\$\text{\$YMigaro}\$\text{\$\text{\$\text{\$\text{\$\text{\$YMigaro}\$\text{\$\$

ただし、このパラメータでは、予約語 \$TARGETDIR \$が指定できない。そのため、コマンドプロンプトから実行できるように、issファイルの参照先は \$TARGETDIR \$が指す場所を記述する。ここでは、C:\Migaro\setuppcE.issとなる。(「2-1. アプリケーション情報」の「ターゲットフォルダ」参照)。【図 19】

以上により、インストール先に、配布 された応答ファイルを参照しながら、サ イレントインストールが行われる。

なお、サイレントインストールで注意 する点として、セットアップの結果が、 応答ファイルと同じ場所に書き込まれ る。具体的には、セットアップの結果ファ イルとして setup.log が作成され、成功 すれば ResultCode=0 と書き込まれる。 このため、配布先は参照だけでなく、書 き込みができる場所でなければならな い。

6.補足および注意点

● InstallAware のインストール

InstallAware は、別途インストール する必要がある。Delphi または RAD Studio の製品 DVD 起動時に表示され るメニューから選択し、インストールを 行う。

ただし、Version2010 以降のメニューからインストールした場合、英語 UI 版となる。日本語 UI 版は、インストールメニューには表示されていないため、直接 DVD にある下記のインストーラを実行する。

¥InstallAware¥ia6-codegear-expressspecial-edition.exe

dbxdrivers.ini

前述の「3. dbExpress でのポイント」で、dbxdrivers.ini の配布先を、プログラムと同じ場所と説明した。しかし、レジストリにその場所を登録すると、以降は EXE と同じ場所に配布する必要はなくなる。

キーは、VersionXE の場合、レジストリHKEY_CURRENT_USER\\$Software\\$Embarcadero\\$BDS\\$8.0\\$DB\Expressのキー「Driver Registry File」に、dbxdrivers.iniが存在するパスを指定する。そして、dbxdrivers.iniの配布先をそのパスとすることで、EXE以外の場所が参照される。

InstallAware では [セットアップ・アーキテクチャ | レジストリ] から指定する。図 20 のように、①で参照元のキーのパスを選択すると、②にキーのDriver Registry File が表示されることを確認し、選択する。③で参照元と同じ

HKEY_CURRENT_USER を選択した 状態で、「値の追加」ボタンを押すと、 図 21 のように参照元のレジストリキー の階層でキーが追加される。【図 20】【図 21】

7.まとめ

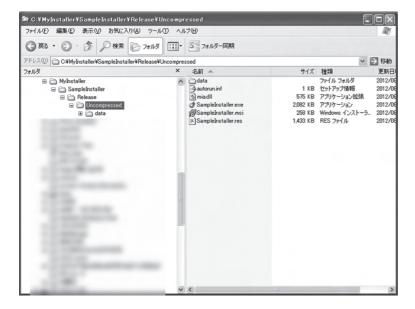
今回は Delphi/400 運用環境に必要なファイルの配布だけでなく、Delphi/400のインストーラを呼び出す方法までを説明した。

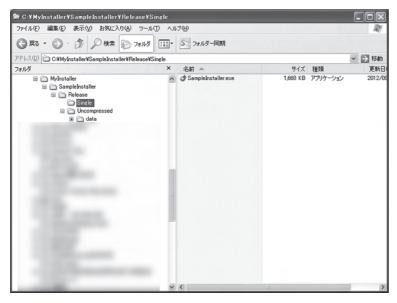
ツリーメニューに従って確認していく と分かるが、今回紹介していない機能も まだ InstallAware には多くあるので、 Delphi/400 の運用環境構築に限らず、 さまざまなファイル配布時に役立ててい ただきたい。

また、InstallAware で行えない細かい 制 御 が 必 要 な 場 合 な ど に は、Delphi/400 で作成したアプリケーションから、InstallAware で作成したインストーラを CreateProcess 等 のWindows API 関数を使って呼び出すことも可能である。

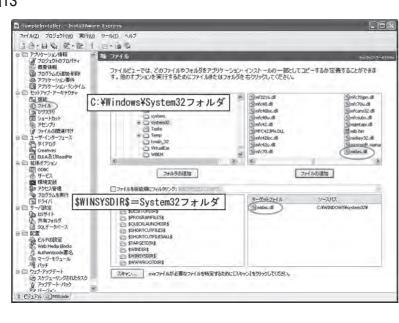
とはいえ、InstallAwareからは、ファイル配布やレジストリキー設定等を画面で存在を確認しながら行えるなど、操作しやすく、機能も充実している。InstallAwareの特性を活かせば、かなり高機能なインストーラを作成できるだろう。

M

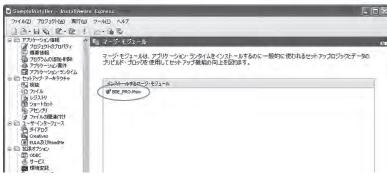


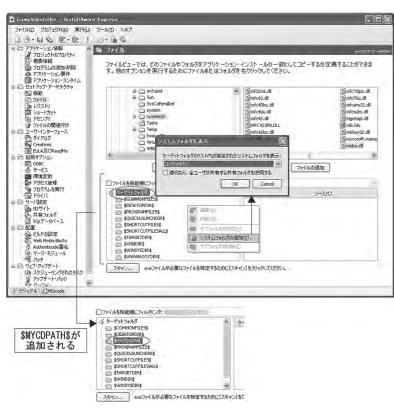












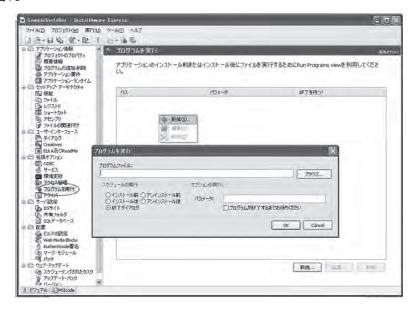
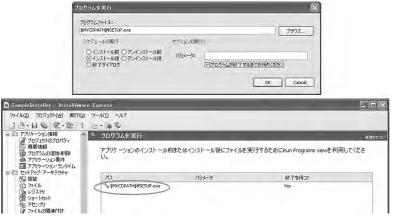


図17



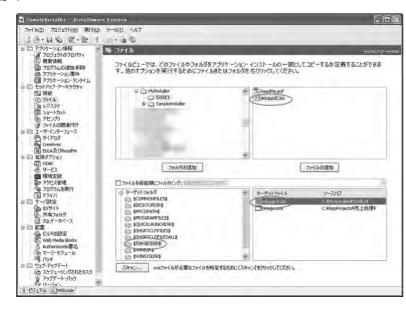
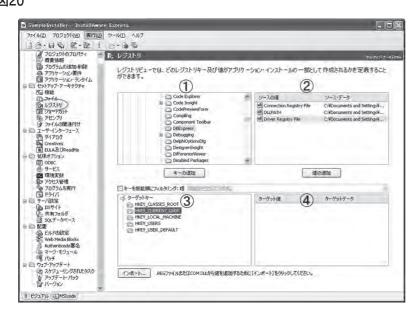
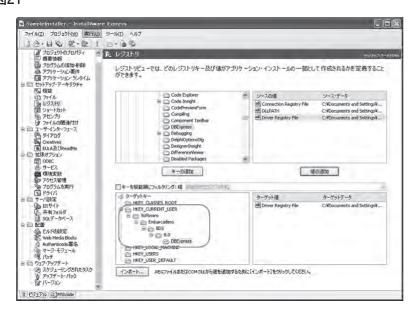




図20





前田 和寬

株式会社ミガロ.

システム事業部 システム2課

カスタマイズコンポーネント入門 一開発効率向上テクニック

コンポーネントのカスタマイズ(継承、機能追加)を用いて、 Delphi/400 の開発効率向上ノウハウを解説する。

- ●オブジェクト指向プログラミング
- ■コンポーネントのカスタマイズ
- ●コンポーネントへの機能追加
- ●まとめ



略歴 1983 年 11 月 21 日生まれ 2006 年大阪工業大学情報科学部卒 2006 年 04 月株式会社ミガロ. 入社 2006 年 04 月システム事業部配属

現在の仕事内容 Delphi/400を利用した、システム の受託開発および保守対応などを担 当している。

1. オブジェクト指向 プログラミング

システムの構築・運用において、開発 効率の向上を図ることは非常に重要であ る。プログラミング/メンテナンスの効 率化を目的として、Delphi 言語を選択 する開発者も多いのではないだろうか。 Delphi 言語の特徴はコンポーネントパ レット、オブジェクトインスペクタを中 心としたビジュアル開発ができることだ が、本質はオブジェクト指向を取り入れ ている点にある。

オブジェクト指向プログラミングとは、オブジェクトを中心に考えていくプログラミング手法のことだ。オブジェクト指向プログラミング言語では、その言語が備えるクラスと継承の仕組みを利用すると、開発効率が格段に向上する。

Delphi/400 ももちろんクラスと継承 の仕組みを備えており、その代表的なク ラス継承は以下である。

・コンポーネントの継承(以下、カスタ

マイズとする)

・フォームの継承

今回は、上記のうち、コンポーネント の継承、カスタマイズを行うと、どのよ うに開発効率が向上するのか考えてみよ う。

通常、システムを運用・稼働していると、エンドユーザーから変更要望が出てくる。例えば、「照会画面でデータの並び順を変更したい」という要望があがったとする。この場合、対象画面に「データの並び順を変更する」処理を追加すればよい。しかし、対象が1画面から10画面に増えると、それに伴い開発工数も増えてしまう。一方、コンポーネントをカスタマイズする場合、対象画面が増えても開発工数は画面数に依存しない。また、コンポーネントで処理記述を一元管理できるため、画面ごとの動作検証も必要最低限で済む。

本稿では、コンポーネントのカスタマイズにより、Delphi/400の開発効率を向上させる方法を紹介したい。

なお、フォームの継承方法については 『ミガロ. テクニカルレポートNo.1 2008 秋』のレポート「フォーム継承による効 率向上開発手法」で分かりやすく紹介し ているので、ぜひ参考にしていただきた い。

2.コンポーネントの カスタマイズ

コンポーネントの継承の概念について、Delphi/400の開発でよく利用する TEdit と TMaskEdit を例に考えてみよう。

TEdit と TMaskEdit はクラス定義が 異なるが、コンポーネントの機能・性質 についてはそれほど大きく変わらない。 その違いは TMaskEdit のクラス名が示 すとおり、EditMask プロパティが定義 されているくらいだろう。TEdit と TMaskEdit は継承元のコンポーネント が共通のため、機能・性質が非常に近い のである。これらは、コンポーネントの 継承関係を図式化して確認すると非常に

図1 TEditとTMaskEditの継承関係



図2 コンポーネントの新規作成



図3 継承元コンポーネントの指定



分かりやすい。【図1】

●カスタマイズのポイント

カスタマイズを行う際には注意すべき ポイントが存在するので、以下に説明す る。これらのポイントに気を付ければ、 コンポーネントのカスタマイズも容易に できる。

(1) コンポーネントを継承する

カスタマイズはコンポーネントを継承 利用する。つまり、新しい機能・性質を 追加する際、継承元のコンポーネントで 定義されたプロパティ、手続き、関数を 利用できる。

例えば、TMaskEdit をもとに作成したカスタマイズコンポーネントをコンポーネント A、TEdit をもとに作成したカスタマイズコンポーネントをコンポーネント Bとする。コンポーネント A は TMaskEdit を継承しているため、EditMask プロパティを利用できる。それに対して、コンポーネント Bの継承元である TEdit にはもちろん EditMask プロパティが定義されていないため、コンポーネント Bでは EditMask プロパティを利用できない。

(2) 動的なコーディングを行う

コンポーネントは不特定多数のプロジェクトで利用するため、処理記述を動的に行う必要がある。通常、TButtonのActionプロパティにAction1が割り当てられている場合、Action1と指定すればよい。しかし、カスタマイズコンポーネントの処理記述では、同様にActionプロパティの設定内容を指定する場合、自身を基準としてSelf.Actionと動的に表現する(Self は省略可)。

●カスタマイズの手順

ではここから、実際に Delphi/400 でカスタマイズコンポーネントを作成する手順を紹介していこう。今回は、TDBGridを継承して TTRDBGrid というカスタマイズコンポーネントを作成する。なお、図表の開発環境は Delphi/400 Version2010を使用している。

①コンポーネントを新規作成する メニューの [ファイル | 新規作成 | そ の他]を選択する。新規作成ウィンド ウの [Delphi プロジェクト | Delphi ファイル | コンポーネント]を選択する。 【図 2】

②継承元コンポーネントを指定する

VCL コンポーネントの新規作成ウィンドウの第1画面で、コンポーネントのリストから TDBGrid を選択し、[次へ]ボタンを押下する。【図3】

③新規コンポーネントの設定を行う

VCL コンポーネントの新規作成ウィンドウの第2画面で、「クラス名」にTTRDBGrid、「パレットページ名」にTechnicalReport を設定する。

今回のカスタマイズコンポーネントのインストール時、この「パレットページ名」の設定値が、新規追加されるコンポーネントパレットのパレットページ名となる。【図 4】

④ pas ファイル名を指定する

VCL コンポーネントの新規作成ウィンドウの第2画面で、「ユニット名」に TechnicalReportControl.pas を指定する。作成するカスタマイズコンポーネントの pas ファイル名、および、保存先を指定する。今回は、保存先に C:\ Projects\TechnicalReport\Lib を指定する。【図 4】【図 5】

手順①~④により、TDBGridを継承したTTRDBGridというカスタマイズコンポーネントを作成できた。ただし、生成されたソースを見れば分かるように、まだ個別に機能を追加していないため、TTRDBGridはTDBGridと機能的に変わらないテンプレートである点に注意してほしい。【図 6】

作成したコンポーネントを Delphi/400 開発環境へインストールす るには、パッケージファイルが必要とな る。続けて、パッケージファイルを作成 する手順を紹介しよう。

⑤パッケージファイルを新規作成する メニューの[ファイル | 新規作成 | パッケージ]を選択する。Packagel. bpl が新規作成されたのが確認できる。 【図 7】

⑥コンポーネントを追加する

メニューの [プロジェクト | プロジェクトに追加] を選択する。プロジェクトに追加 ウィンドウで、C:\Projects\TechnicalReportControl.pas を指定する。

コンポーネントの追加後、メニューの [ファイル | すべて保存]を選択し、同 じフォルダに TechnicalReport.dproj として保存する。【図 8】

手順⑤⑥により、コンポーネントのインストール用パッケージファイルが作成できた。なお、これまで作成したコンポーネントやパッケージファイルは配布を行うことで、他のユーザーが Delphi/400 開発環境へインストールして共有することができる。

最後に、パッケージファイルを使って、 コンポーネントをインストールする手順 を紹介しよう。

⑦パッケージファイルを開く

メニューの[ファイル | 開く]を選択 する。プロジェクトを開くウィンドウで TechnicalReport.dproj を選択し、パッ ケージファイルを開く。

⑧コンポーネントをインストールする メニューの「プロジェクト | プロジェ クト名を再構築]を選択する。再構築後、 プロジェクトマネージャの TechnicalReport.bplを右クリックする。表示メニューの「インストール」を 選択する。

インストール後は図のように、 TTRDBGridを正常にインストールで きたことを確認できる。【図 9】

⑨ Delphi/400 開発環境のライブラリパスの設定を行う

メニューの [ツール | オプション] を 選択する。オプションウィンドウの [環 境オプション | Delphi オプション | ラ イブラリ Win32 | ディレクトリ | ライ ブラリパス] の […] ボタンを押下する。 ディレクトリウィンドウで C:\Projects\ TechnicalReport\Lib を指定する。【図 10】

インストールしたコンポーネントを開発で利用する場合、Delphi/400 開発環

図4 新規コンポーネントの設定



図5 pasファイルの指定

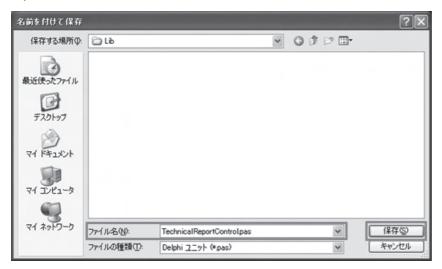


図6 作成されたTTRDBGrid



境にパッケージファイルの格納フォルダ をライブラリパスとして登録する必要が ある。

以上で、カスタマイズコンポーネント を作成し、その作成したカスタマイズコ ンポーネントを、Delphi/400 開発環境 にインストールできた。Delphi/400の 再起動後、カスタマイズコンポーネント TTRDBGrid が利用可能となる。

3.コンポーネントへの 機能追加

本章では、カスタマイズコンポーネン トへの機能追加を紹介したい。今回は2 章で作成した TTRDBGrid に「データの 並び順を変更する」機能を追加してみよ

【図 11】のような TTRDBGrid を利用 している照会画面で、明細部のタイトル をダブルクリックするごとに、選択列を 基準として並び順を変更できるようにし よう。【図 11】

データの並び順は「昇順設定→降順設 定→設定なし→昇順設定→…」、と設定が 切り替わるようにする。また、並び順の 設定が分かりやすいように、選択列のタ イトルのフォント色も変更する。以下の ような切り替えになる。

昇順設定:赤色

降順設定:青色

設定なし: 黒色

昇順設定:赤色

1

●データの並び順

処理を実装する前に、データの並び順 を変更する方法を説明したい。

データの並び順は、TTRDBGrid に紐 付くデータセットのインデックスを利用 する。照会画面の作成時、TTableや TQuery を利用することがよくある。し かし、並び順の変更はローカルキャッシュ に取り込んだデータのインデックスを利 用したいので、今回はデータセットに TClientDataSet を前提とする。

次に、TClientDataSetのインデックス

設定について説明する。インデックス設 定には、IndexName と IndexFieldNames の2種類のプロパティが存在する。前者 の IndexName は、Index 定義(フィール ドの優先順位、フィールドごとの昇順/ 降順を指定したもの)をあらかじめ準備 して設定する必要がある。後者の IndexFieldNames は、フィールドの優先 順位は指定できるが、降順の指定はで きない。今回は昇順/降順を選択で きるようにしたいので、Index 定 義とIndexNameを利用して並び順 を変更する。なお、IndexName と IndexFieldNames は互いに排他関係にあ るため、どちらか一方しか設定できない。

●機能追加

では、機能追加に必要な処理記述を以 下に洗い出す。

- ・タイトルをダブルクリックするごとに、 並び順を変更する。
- ・変更前のインデックス設定を退避/復 元する。
- ・選択列のフォント色を変更する。

ここでは、メインの処理である「タイ トルをダブルクリックするごとに、並び 順を変更する」を中心に説明したい。

まず、タイトルのダブルクリックによ り、並び順を変更するため、選択列を特 定する必要がある。WMLButtonDown イ ベントで、マウスカーソル位置から選択 列を取得し、WMLButtonDblClkイベン トにて、インデックスの設定処理を呼び 出す。【ソース 1】 【ソース 2】

次に、特定した選択列をもとに、イン デックスを設定する。ここで「動的なコー ディングを行う」というポイントを思い 出してほしい。

コンポーネントの処理記述時、 TTRDBGrid に紐付くデータセットの Name や選択列に設定されるフィールド の FieldName は不明である。そのため、 自身を基準として、紐付くデータセット は DataSource.DataSet、選択列に設定さ れるフィールドは Columns.Items[FCoord. X-1].FieldName と表現する。

また、今回の Index 定義は昇順/降順 を問わず、GRID_IDXと命名する。 TClientDataSet の AddIndex を利用し、 選択列のフィールドと昇順/降順を設定

する。昇順の場合、[ixCaseInsensitive]、 降順の場合、[ixDescending] を指定する。

後は IndexName に GRID_IDX を設定 することで、並び順の変更ができる。【ソー ス3】

なお、その他の処理の「変更前のイン デックス設定を退避/復元する」 「選択列 のフォント色を変更する」については、 末尾の参考ソースを参照していただきた い。【ソース 4】【ソース 5】【ソース 6】【ソー ス7]

では、コンポーネントへの機能追加が できたので、コンポーネントの再インス トール後に照会画面の動作を確認してみ よう。明細部のタイトルをダブルクリッ クするごとに、昇順や降順に明細データ の並び順が切り替わることが確認できる だろう。また、同様に TTRDBGrid を利 用している画面が他にもあれば、すべて の画面で並び順の変更が可能になる。【図 12】【図 13】

4.まとめ

コンポーネントのカスタマイズと聞く と難しいイメージを抱くかもしれない が、処理手順を見ていただいたとおり、 カスタマイズコンポーネントのテンプ レートは簡単な操作で作成できる。また、 コンポーネントへの機能追加についても ポイントを押さえれば、比較的簡単に実 現できる。

一度、カスタマイズコンポーネントを 作成・機能追加すれば、そのコンポーネ ントを利用しているすべての画面で、追 加した機能が利用できるようになる。ま た、作成したコンポーネントは、他のユー ザーもインストールして利用が可能だ。 このようにコンポーネントのカスタマイ ズにより、Delphi/400の開発効率を向 上できることが理解いただけただろう。

本稿で紹介した TTRDBGrid の「デー タの並び順を変更する」機能を参考にし て、コンポーネントのカスタマイズ、お よび独自の機能追加にぜひ挑戦してほし い。これらの技術情報がシステム開発の 助けとなれば幸いである。

M

図7 パッケージファイルの新規作成

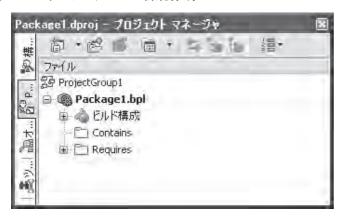


図8 コンポーネントの追加と保存

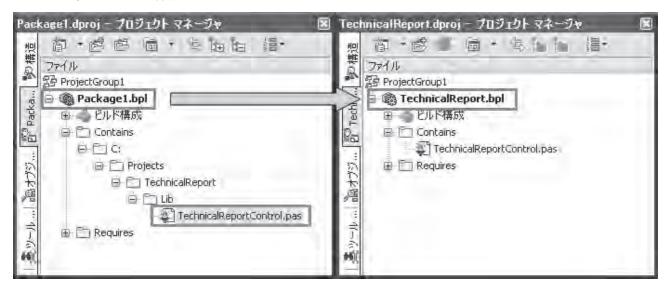


図9 コンポーネントのインストール



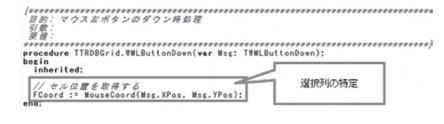
図10 ライブラリパスの設定



図11 照会画面と追加機能



ソース1 WMLButtonDownイベント



ソース2 WMLButtonDblClkイベント

ソース3 Sorting手続き

```
目的 ソート都運
引動: ソート種別(1:異類、2:簡単、0:解除)
実養:
procedure TTROBGrid. Sorting (ASortKind: Integer);
   AbataSet; TClientDataSet:
begin
inherited:
                                                                            TTRDBGridを基点とした動的表現
   | TClientDataSetが設定されている。かつ、
| 特定別のフィールド=対策項目の場合。
| If (Assigned(DataSource)) and (Assigned(DataSource, DataSet)) and (DataSource, DataSet, Active) and (DataSource, DataSet is TClientDataSet) and (FCoord, X >= 1) and (FCoord, X <= Columns, Count) and (Columns, Items[FCoord, X-1], Field, FieldWind <> (KCalculated) then hersin
   begin
ADalaSet := (DataSource.DeløSet am TCfientDateSet);
      // 本来の設定の過避
If (FiNowSortKind = 8) then
      begin
Fs8aseIndexName := ADataSet.IndexName;
Fs8aseIndexFieldNames := ADataSet.IndexFieldNamem
                                                                                             変更前インデックスの退避
        Indexの初期化
      try ADataSet.DeleteIndex(GRID_IDX);
      except
      end;
                                                                           TTRDBGrioを基点とした動的表現
      If (ASortKind : 1) Lhen
      begin
ADataSet.AddIndex(GRID_IDX, Columns.Items[FCoord.X-1].FieldName, [ixGaseInsensitive]);
ADataSet.IndexName := GRID_IDX;
         FiNowSortKind := 1;
FaNowIndexField := Columns.Items[FCoord.N-11.FieldName;
      // 2:跨廊
else if (ASortKind = 2) then
begin
ADataSet.AddIndex(GRID_IDX, Calumns.Items[FCoord.X-1].FieldName. [ixDescending]);
ADataSet.IndexName := GRID_IDX;
      FiNowSortKind := 2;
FaNowIndexField := Columns.Items[FCoord.X-1].FieldName;
end
      else if (ASortKind = 0) Lhen
begin
ADataSet.IndexName := FsBaseIndexName;
ADataSel.IndexFieldNames := FsBaseIndexFieldNames;
      FiNowSortKind := 0:
FaNowIndexField := ':;
end;
      ADataSet.First:
end:
```

参考ソース4 TTRDBGridの宣言部

参考ソース5 BeforeClose手続き

参考ソース6 Loadedイベント

```
| Pasta Pa
```

参考ソース7 DrawCellイベント

```
/4--
 ****************************
 procedure TTROBGrid.DrawEell(ACol. ARow: Integer: ARect: TRect;
    AState: TGridDrawState);
Const

// またセンター表示とする
AlignFlass: acray [TAligneent] of Integer =
( DI CENTER or DI WORDBREAK or DI EXPANDIABS or DI NOPREFIX,
DI CENTER or DI WORDBREAK or DI EXPANDIABS or DI NOPREFIX,
DI CENTER or DI WORDBREAK or DI EXPANDIABS or DI NOPREFIX):
 var
sText: String;
Flags: LongInt;
 C: integer:
DrawFlg: longint:
Cotumn: TCotumn:
begin
     inherited DrawCell(ACol, ARow, ARect, AState);
    if (ARow = D) and (ACol ◇ D) then begin 舞舞の意見語し InflateRect(ARect. -1. -1); Canvas.Brush.Color: FixedColor; Cenvas.Full.Assign(TitleFant);
       // フォント島の設定
if (Assigned(DataSource)) and
(Assigned(DataSource.DataSet)) and
(DataSource.DataSet is TClientDataSet) them
         パソート設定を行って()る場合。
if ((FiNowSortKind = 1) or (FiNowSortKind = 2)) and
(Columns.Items[ACol-1].FieldName = FsNowIndexField) them
               ア/年順ソート
If (FiNowSortKind = 1) then
begin
               Canvas.Font.Color := ciRed;
              //## y - k
else if (FiNovSortKind = 2) then
begin
Canvas,Font,Color := ciBlue;
end;
                                                                                              選択列のフォント色を設定
// 文字の再接面
sText ;= Self.Columns[ACol-1].Title.Caption;
Flags := DT_SINGLELINE or DT_VOENTER or AllgnFlags[Columns[ACol-1].Title.Allgnment];
DrawText(Canvas.Handle, PChar(sText), Length(sText), ARect, Flags);
end;
```

図12 照会画面・会社コードの昇順設定



図13 照会画面・会社コードの降順設定



吉原 泰介

株式会社ミガロ.

RAD事業部 技術支援課 顧客サポート

Delphi/400活用! スマートデバイスアプリケーション開発

拡張性の高い Delphi/400 ならでは。「jQueryMobile」を使用し、 見栄えのよいスマート画面開発にチャレンジしよう。

- ●スマートデバイスでのWebアプリケーション活用
- ●Delphi/400のWebアプリケーション開発
- ●jQueryMobileを利用したスマートデバイス開発
- ●まとめ



略歴 1978年03月26日生 2001年龍谷大学法学部卒 2005年07月株式会社ミガロ. 入社 2005年07月システム事業部配属 2007年04月RAD事業部配属

現在の仕事内容 Delphi/400 と JC/400 の 製 品 試 験、および月 100 件に及ぶ問い合 わせやサポート、セミナー講師など を担当している。

1. スマートデバイスでの Webアプリケーション 活用

ここ1、2年でiPhone やiPad をはじめ、スマートデバイス端末が急速に普及している。これらスマートデバイス端末は個人の利用だけでなく、ビジネスツールとして、すでに導入・利用を始めている企業も少なくない。そうした背景もあり、今回は、Delphi/400でスマートデバイス向けのアプリケーションを開発するテクニックを紹介したいと考えた。これから紹介するテクニックを使えば、図1のような画面スタイルのスマートデバイスアプリケーションが実現できる。【図1】

Delphi/400 では C/S アプリケーションだけなく、Web アプリケーションも 開発できる。つまり、Web アプリケーションを開発すれば、スマートデバイス上の Web ブラウザから IBM i のデータを扱うことができるようになる。

特に企業で使用するスマートデバイス

のシステム開発は昨今、端末にインストールするアプリケーションよりも、Webアプリケーションが主流になってきている。その理由としては大きく3つある。

●アプリケーションプログラムの統一

1つ目の理由は、アプリケーションプログラムを統一できる点である。

スマートデバイスとは言っても、iPhone や Android 等の OS やタブレットの規格の違いなど、端末はさまざまな種類が使われている。決められた端末(プラットフォーム)向けに開発されたアプリケーションは、異なる機種の端末では動作させることができない。これはWindowsのアプリケーションが、Macintosh上で動作しないのと同様である。そのため、同じアプリケーションであっても、iPhone 用、Android 用など機種が異なれば、それぞれにあった複数のアプリケーション開発が必要になってくる。

しかし Web アプリケーションであれ

ば、Web ブラウザ上で動作するため、Web ブラウザが使える環境であれば共通して使用することができる。これが大きな強みとなる。

例えば、iPhone であっても Android であっても同じ Web アプリケーションを使うことができるので、シンプルに 1 つの開発だけで統一できる。もちろん、Web アプリケーションであれば、スマートデバイスだけでなく、Windows や Mac、Linux といった PC 上にも展開し、利用することができる。

●アプリケーションの配布・運用

2つ目の理由は、開発したアプリケーションの配布・運用面である。

通常、スマートデバイスで利用するアプリケーションは、iPhone であってもAndroid であっても、開発したアプリケーションをマーケット上で公開したり、パッケージを配布したりする必要がある。しかし、企業で使用するアプリケーションをマーケット上で公開することは、難しいだろう。またiPhoneでは、

Apple のマーケット公開の審査も厳しいため、いろいろと準備が必要になって くる。

その点、Web アプリケーションであれば、Web サーバにアプリケーションを用意すれば、ブラウザで URL を指定するだけで利用できる。上記のような配布や運用に苦慮する必要もない。

●セキュリティ

3つ目の理由はセキュリティである。

スマートデバイスは手軽に携帯して利用できる半面、紛失などの可能性を十分に考慮する必要がある。特に、スマートデバイス上にインストールして動作するアプリケーションが IBM i のデータを利用しているならば、接続手段や接続情報を端末内に持っているということである。つまり、スマートデバイスを紛失した場合や、ウイルスなどで不正アクセスされてしまった場合には、社内 IBM i への接続情報や設定が漏洩する恐れが生じる。

この点においても、Web アプリケーションは端末上にアプリケーションの情報を持たず、あくまで Web サーバ上の URL にアクセスして利用するため、セキュリティ的にも安心して利用することが可能だ。

本稿では、こうした優れた特徴を持つスマートデバイス向けのWebアプリケーションを、Delphi/400で開発する方法を説明する。ただし、Delphi/400の開発手法は1つではなく、さまざまな手法で開発が可能である。今回はその一例の紹介であることをご了承いただきたい。

2.Delphi/400のWeb アプリケーション開発

Delphi/400 では、Web アプリケーション開発のために「VCL for the Web (旧称 IntraWeb)」と呼ばれる機能が用意されている。この VCL for the Web では、C/S アプリケーションと同じ開発手法で Web アプリケーションが開発できる。Delphi/400 開発者によっては非常に便利な機能である。

最初に、Delphi/400のWebアプリケーション開発についての概略を説明

する。 なお 今 回 は、Delphi/400 VersionXE のバージョンを使用している。

【Web アプリケーションの開発手順】

- (1) プロジェクトの作成
- (2) コンポーネントの配置
- (3) プログラムの作成
- (4) コンパイルと実行

2-1. プロジェクトの作成

新規に Web アプリケーションのプロジェクトを作成するには、[ファイル]のメニューから [新規作成 | その他]を選ぶと、ダイアログが表示される。 [Delphi/400 プロジェクト | VCL for the Web] より VCL for the Web Application Wizard を起動する。【図2】

Wizard のダイアログが表示されるので、ここでは ISAPIExtension を選択して進める。【図 3】

これで、Web用の新規プロジェクト が作成できた。【図 4】

2-2. コンポーネントの配置

Delphi/400 では、Web アプリケーション用に、C/S アプリケーションと 類似のコンポーネントが用意されている ので、同じ感覚でプログラムを作成する ことができる。

例えば、TIWButton、TIWRadioGroup、 TIWDBGrid で画面設計をして、簡単な 照会画面を作成してみる。【図 5】

2-3. プログラムの作成

IBM i のデータは、C/Sアプリケーションと同じコンポーネントで扱うことができる。また今回は、dbExpress 接続を使用する。データの表示はTClientDataSetコンポーネントを使い、次のように構成する。【図 6】

TSQLConnection

- →TSQLQuery
- →TDataSetProvider
- →TClientDataSet
- →TDataSource
- →TIWDBGrid (画面表示)

イベントプログラムはボタンを押下時 (OnClick) に、TIWRadioGroup の指 定に従って、TClientDataSet を Open するロジックとする。【ソース 1】

2-4. コンパイルと実行

プログラムのコンパイルを行うと、dll のアプリケーションが作成される。Windows サーバ上で稼働させた IIS (Internet Information Service) に作成した dll を配置して、Web ブラウザから実行してみよう。

PCのWebブラウザはもちろん、スマートフォンなどのWebブラウザからもアプリケーションを実行して、IBMiのデータを活用することができるのである。今回はスマートデバイスに画面サイズをあわせているため、PCブラウザでは若干画面サイズを小さくしている。【図7】

ここまで、Web アプリケーションの 開発手順の概略を説明した。

なお、開発の詳しい手順については、 『ミガロ. テクニカルレポート No.2 2009 年秋』掲載のレポート「Delphi/400 を 利用したはじめての Web 開発」で分か りやすく解説しているので、参考にして いただきたい。

3.jQueryMobileを 利用したスマート デバイス開発

さて、前章で Delphi/400 での Web アプリケーション開発手順を紹介したが、スマートデバイスで使うにしては、画面デザインが少々物足りないと感じる方もいるだろう。 Web アプリケーションの見た目をよくするためには、画面用にデザイン画像を用意したり、JavaScript やmeta タグ等を組み込んだりすることで工夫できる。

今回は、Web アプリケーションをスマートデバイスで見栄えよく完成させるために、「jQueryMobile」というオープンソースを活用する手法を紹介する。

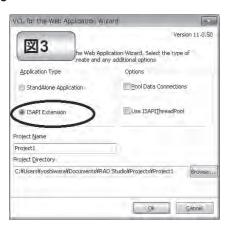
3-1. jQueryMobileとは?

まず、jQueryMobile について、説明したい。【図 8】



図2





jQueryMobile はスマートフォンやタブレットで使用する Web アプリケーション用にユーザーインターフェースの画面デザインや部品を提供してくれるフレームワークである。jQueryMobile のオリジナルは海外サイトで公開されている。もちろん、日本語に訳されたリファレンスのサイトもあるので安心していただきたい。【図 9】

[jQueryMobile]

http://jquerymobile.com

【jQuery Mobile 1.1.0 日本語リファレンス】

http://dev.screw-axis.com/doc/jquery_mobile/

使い方は非常に簡単で、公開されている html ソースを組み込むだけで利用することもできる。実は、Delphi/400 の「VCL for the Web」 には html を組み込む機能が用意されているので、この jQueryMobile をそのまま利用することができるのである。

3-2. htmlの組み込み機能

早速 jQueryMobile を Delphi/400 で利用するために、html を組み込む方法を説明しよう。

まず、使用するのは TIWTemplate ProcessorHTML コンポーネントである。このコンポーネントは、html を指定して、画面デザインを置き換えることができる。コンポーネントの使い方は次のとおりである。

- ① Web アプリケーションを配置するフォルダ内に templates というフォルダを新規作成して、利用したい画面 html を保存する。注意点としては、html ファイルの文字コードは必ず UTF-8 で保存する必要がある。【図 10】
- ② TIWTemplateProcessorHTML コンポーネントの Templates プロパティの Default に、①で保存した画面 html 名を設定する。【図 11】
- ③ Web アプリケーションの画面フォーム に LayoutMgr プロパティがあるので、そ

こに TIWTemplateProcessorHTML コンポーネントを設定する。【図 12】

これだけの簡単な作業で、Delphi/400 アプリケーションの画面に html を利用す ることができるようになる。

3-3. 画面レイアウトの作成

それでは、jQueryMobile を利用して画面デザインを作成してみよう。jQueryMobile のサイト上に「単一ページテンプレート」が用意されているので、これを利用する。

まず、コピーした html を templates フォルダに保存する。【ソース 2】

【参考 URL】

http://dev.screw-axis.com/doc/jquery_ mobile/#/doc/jquery_mobile/ components/pages_dialogs/single_ page_template/

続いて、TIWTemplateProcessorHTML コンポーネントに html を設定して、実際 にアプリケーションを実行してみよう。 実行すると、今まで作成していた画面(図 7)とは、全く異なる html が表示される ことが確認できるだろう。【図 13】

ところで、html をうまく組み込むことができたが、画面は jQueryMobile サイトからコピーしたテンプレートの内容のままだ。そこで、html のページタイトル、コンテンツ、フッターを書き換えてみよう。【ソース 3】

再度実行してアプリケーション画面を確認すると、書き換えた html が反映されている。【図 14】

ちなみに、html は Delphi/400 の開発 環境で開くこともできる。つまり、 Delphi/400 ソースと同じように、html を 編集することができる。【図 15】

以上で、スマートデバイスらしい画面 デザインを取り込むことができた。さら に見た目だけでなく実は、この画面はス マートデバイスの端末の形態にあわせて、 画面を自動調整してくれるのである。もち ろん縦、横の画面回転にも対応している。

3-4. コンポーネントとhtmlの連携

html で、スマートデバイス用の画面デ

ザインを適用することができた。ただし、 今の状態では、Delphi/400 で作成したコ ンポーネントが表示されていない。ここ からは、コンポーネントを html と連携さ せる方法を説明していく。

連携させる方法は簡単である。html に、 コンポーネントの名前を次のように追記 すればよい。

|%コンポーネント名% |

では、実際に、html のコンテンツ部分 にコンポーネント名を組み込んでみよう。

コンパイルして実行すると、htmlの画面の中にコンポーネントが組み込まれ、表示されていることが確認できる。つまり、画面だけ html で表示して、裏ではDelphi/400のプログラムがそのまま動作しているという仕組みである。【ソース4】【図 16】【図 17】

3-5. ラジオグループのカスタマイズ

さて、コンポーネントが表示されたの で、ここからはコンポーネントの見た目 についても、jQueryMobile を活用してカ スタマイズしていく。

まず TIWRadioGroup については、 jQueryMobile のサイト上にラジオボタン のサンプルが用意されている。このソー スをコピーして、部品の名前を TIWRadioGroupのコンポーネント名 (Nameプロパティ値)に変更し、html を書き換えよう。【ソース5】

【参考 URL】

http://dev.screw-axis.com/doc/jquery_mobile/#/doc/jquery_mobile/components/forms/radiobuttons/

アプリケーションを実行して画面を表示すると、ラジオグループがカスタマイズされていることが確認できる。【図 18】

3-6. グリッドのカスタマイズ

次に、データを表示しているグリッド をカスタマイズする。グリッドについて は、jQueryMobile のサイト上にリスト ビューとしてサンプルが用意されている。

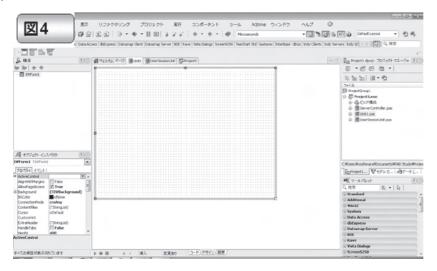
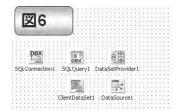


図5



図6



ソース1

yース1:表示ボタン押下時 Delphi/400 procedure TIMForm1.IMButton1Click(Sender: TObject); begin //データを閉じる ClientDataSet1.Close; //SQLのパインド変数に画面の指定をセット(ここでは区分) SQLのパインド変数に画面の指定をセット(ここでは区分) ClientDataSet1.Open; end;

【参考 URL】

http://dev.screw-axis.com/doc/jquery_ mobile/#/doc/jquery_mobile/ components/lists/docs/

この html を参考にして、TIWDBGrid をカスタマイズできる。

ただし、データの量によっては、可変で変更されるリストビューを html で事前に定義することが難しい。そのため、TIWDBGrid に関しては、Delphi/400のプログラムの中で動的に html を作成することにする。

●動的なリストビュー

まず、ClientDataSet のデータ量によって、リストビューの html ソースを動的に 作成する関数を用意する。【ソース 6】

そして、html 上で {% IWDBGrid1 %} と記述している部分を、{% ListView%} へ変更する。これによって、TIWDBGrid を表示せずに、新しく ListView の描画が行えるようになる。このリストビューは、TIWTemplateProcessorHTML コンポーネントの UnknownTag イベントで、関数を呼び出して作成している。【ソース7】 アプリケーションを実行して、表示ボタンをタッチしてみよう。すると、データの表示がグリッドではなく、リストビュー形式で表示される。【図 19】

ここまでで、元の画面部品を全てス マートデバイス向けにカスタマイズする ことができた。

最後におまけとして、リストビューに 画像表示も組み込んでみよう。

フィールドの値と一致する画像ファイルがWebサーバに保存されていることが前提とするが、タグのhtmlコードを追加すれば、簡単にリストビュー上に画像を組み込むことができる。【ソース8】これで画像付きの照会アプリケーションの完成である。【図20】

4.まとめ

さて本稿では、Delphi/400が html を組み込めるという機能を着眼点として、スマートデバイス向けの Web アプリケーション開発方法を説明してきた。拡張性が高い Delphi/400 ならではのスマートデバイス対応である。

今回紹介したjQueryMobileをうまく活用すれば、誰でも容易に高機能なスマートデバイス画面を開発できることをお伝えできただろうか。もちろんhtmlを組み込まなくとも、Webアプリケーションを開発することはできるので、あくまでスマートデバイスに簡単に対応するための1つの手法と考えていただきたい。

また Delphi/400 では、jQueryMobile に限らず、公開されているさまざまな html や JavaScript の技術を利用することが可能だ。ぜひ Web アプリケーション開発の際には、見栄えのよいスマートデバイス画面にチャレンジしてほしい。その際、本稿のテクニックが参考になれば幸いである。

M



図8







図11



図12



ソース2

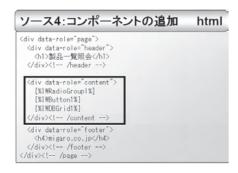


ソース3

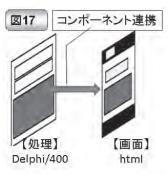




ソース4







ソース5



ソース6

```
yース6 リストビューの動的作成関数

//宣言 public function Render IMGrid: String: end;
//実装部 function IIMForm1.Render IMGrid: String; var
ul: TStringList; begin
//グリッドのhtml文を動的に作成する。
ul:: TStringList.Create; ul.Add('〈ul data-role="listview"〉');
//ClientDataSetのデータ分、リストビューのレコードを作成する。
while not(ClientDataSet1.Eof) do
begin
ul.Add('〈hâ〉' + ClientDataSet1.FieldByName( '製品コード').AsString );
ul.Add('(か高な' + ClientDataSet1.FieldByName( '製品コード').AsString + '〈/p〉');
ul.Add('〈p〉存作: ' + ClientDataSet1.FieldByName( '福格').AsString + '〈/p〉');
ul.Add('〈p〉存作: ' + ClientDataSet1.FieldByName( '在庫数').AsString + '〈/p〉');
ul.Add('〈/ul>'):
(ClientDataSet1.Next; end;
ul.Add('〈/ul>'):
//作成したhtml分を関数から返却する。
Result:= ul.Text;
FreeAndNil(ul); end;
```

ソース7

```
yース7:リストビューの動的作成

procedure TIMForm1.IMTemplateProcessorHTML1UnknownTag(const AName: string; var VValue: string); begin
//html上で(乳istView3)を描画する際に、データがオープンされていれば
//リストビューを動的に作成する。
if ((AName = 'ListView') and (ClientDataSet1.Active)) then
begin
VValue := RenderIMGrid; end; end;
```



ソース8

```
「リース8:画像データの組み込み Delphi/400

function TIWForm1.RenderIWGrid: String:
var
ul: TStringList;
begin

while not(ClientDataSet1.Eof) do
begin
ul.Add('');
ul.Add('');
ul.Add('');
ul.Add('');
ul.Add('
ul.Add('</
```



尾崎 浩司

株式会社ミガロ.

システム事業部 プロジェクト推進室

DataSnapを使用した3層アプリケーション構築技法

多層型アプリケーションのためのフレームワーク「DataSnap」。 これを活用し、Delphi/400 のスキルだけで構築を実現する。

- ●3層アプリケーションとは
- Data Snap とは
- ●DataSnapサーバプログラム作成手順
- ●DataSnapクライアントプログラム作成手順
- ●Windowsサービスを使用したDataSnapサーバの作成
- ●サーバメソッドを呼び出すクライアントプログラムの作成
- ●最後に



略歴 1973 年 08 月 16 日生 1996 年三重大学工学部卒 1999 年 10 月株式会社ミガロ. 入社 1999 年 10 月システム事業部配属

現在の仕事内容 ミガロ. 入社以来、主に Delphi/400 を利用した受託開発を担当している。

1.3層アプリケーション とは

一般的にクライアントサーバ型アプリケーションというと、画面処理+ビジネスロジックを担当する「クライアント(アプリケーション層)」とデータを保持する「データサーバ(データ層)」の2階層で構築することが多いだろう。

対して、3層アプリケーションとは、画面処理を担当する「クライアント(プレゼンテーション層)」、ビジネスロジックを担当する「アプリケーションサーバ(アプリケーション層)」およびデータを保持する「データサーバ(データ層)」の3階層に分割して構築するクライアントサーバ型アプリケーションのことを指す。【図1】

3階層の特徴は、ビジネスロジック部分をアプリケーションサーバ上のプログラムとして、クライアントPC上で実行される画面処理部分のプログラムと完全に分離することにある。

3層アプリケーションとしてシステム

を構築するメリットとしては、次のよう なことが挙げられる。

- (1) ビジネスロジックの変更が行いやす
- (2) 大量データを取り扱うアプリケー ションが構築しやすい
- (3) クライアント PC の環境構築がシン プルになる
- (1) 2層アプリケーションのケースでは、ビジネスロジックがクライアントPC上にあるため、仕様変更が必要となった場合、都度クライアントPCへ最新モジュールの再配布が必要となる。

対して、3層アプリケーションとしたケースでは、ビジネスロジックがアプリケーションサーバ上にあるプログラムとなるため、画面処理部分の変更がない限り最新モジュールはアプリケーションサーバにのみ適用すればよい。

(2) 3層アプリケーションとしたケースでは、ビジネスロジックを記述したプロ

グラムとデータベースとのやり取りが サーバ間でのネットワーク通信となり、 サーバ⇔クライアント間のやり取りは、 画面処理に必要な情報のみに絞り込むこ とができる。

これは、ビジネスロジックが大量の データを処理する必要がある場合に、遠 隔拠点に配置したクライアントPCから 実行指示を行う際の、処理レスポンス向 上に大きく寄与するだろう。

(3) 通常2層アプリケーションのケースでは、クライアントPC上にデータベース通信用の専用モジュールをセットアップしなければいけないが、3層アプリケーションとしたケースでは、クライアントPCは直接データベースを参照しないため、クライアントPCにデータベース通信用モジュールが不要となる。

よって、Delphi/400 の特徴であるランタイム不要な実行モジュールが作成できる点を最大限に活用することが可能になる。つまり、クライアント PC には、実行モジュール (Exe) だけ配布すれば

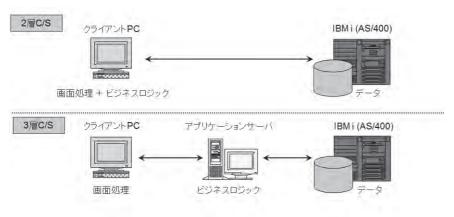
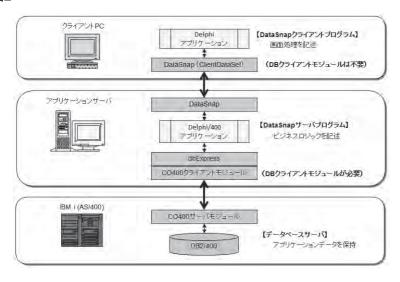


図2



ソース1



よいのである。これは、OS/400のバージョンアップや Delphi/400 運用版ミドルウェアのバージョンアップをより容易にするだろう。

2.DataSnapとは

こういったメリットを持つ3層アプリケーションだが、あまり利用されていないのはなぜだろうか。それは、アプリケーションサーバとクライアントPCとの間の通信を実装する手順が容易ではないからである。

技術的には「CORBA」と呼ばれる分散オブジェクト技術の仕様があるのだが、これをアプリケーションに導入しようとすると、どうしても CORBA に関する専門知識が不可欠となる。つまり、実現は可能だが敷居が高いというのが本音だろう。

ところが、Delphi/400には、「DataSnap」と呼ばれる多層型アプリケーションを構築するためのフレームワークが搭載されている。この DataSnap を利用すれば、Delphi/400 のスキルだけで 3 層アプリケーションが構築できるのである。

DataSnapを使用して構築する3層アプリケーションは、図2のような構成となる【図2】。一見すると複雑そうなイメージではあるが、これらは、通常のアプリケーション同様、コンポーネントを使用したビジュアル開発で作成可能である。

特に、Delphi/400 VersionXEでは、 便利なウィザードが使用できるため、よ り容易に作成できるようになっている。 今回は、この DataSnapを使用した3層 アプリケーションの構築技法を紹介しよ うと思う。

なお、今回作成するサンプルプログラムは、ソース1のDDSより生成された「得意先マスタ(ファイル名:DTMSCP)」を使用する。【ソース1】

3.DataSnapサーバ プログラム作成手順

アプリケーションサーバに配置する、 DataSnap サーバプログラムの作成手順を 見ていこう。

まず Delphi/400 を起動したら、[ファイル | 新規作成 | その他] より新規作成

メニューを開き、「DataSnap Server」を 選択する。【図 3】

するとウィザードが始まるので、次のとおり指定していこう。

- ①第1画面の「プロジェクトの種類」では通常の「VCLフォームアプリケーション」を選択する。今回はサーバプログラムも通常のフォームアプリケーションとして作成する。【図4】
- ②第2画面目の「サーバの機能」画面では、使用するサーバ機能を選択する。ここでは初期値のまま「プロトコル・TCP/IP」「サーバメソッドクラス」にチェックを付けておこう。もし通信手段にHTTPを使用する場合は、ここで選択すればよい。【図5】
- ③第3画面目は、通信に使用するポートを選択する。ここも通常は初期値である「211」を指定しておけばよいだろう。もし同じサーバ上に複数の DataSnapサーバプログラムを配置するような場合には、それぞれのプログラムで異なるポート番号を指定すればよい。【図6】
- ④最後の第4画面目の「サーバメソッド クラスの上位クラス」では、メソッド 等を定義する親クラスを指定する。通 常ここでは TDSServerModule を選択 しよう。これを選択すると、2層アプ リケーション開発においても多用する データモジュールと同様の開発手順が 可能となる。【図7】

以上で、設定が完了である。

ウィザードが完了すると、次の3つの ユニットを持つプロジェクトが生成され る。

- (1) ServerControllerUnit1.pas (DataSnap 本体)
- (2) ServerMethodsUnit1.pas (サーバ実 装用モジュール)
- (3) Unit1.pas (メインフォーム)

プロジェクトが生成されたら、ビジネスロジックにあたる処理を(2)の ServerMethods Unit1.pas にある TServerMethods1 に作成すればよい。ここでは、得意先マスタの内容を取得でき

るサーバを作成していこう。

ServerMethodsUnit1.pas のデザイナを立ち上げ、TSQLConnection、TSQLTable、TDataSetProviderを貼り付けて、データベース接続設定を行ってみよう。なお、この手順は、データモジュールにdbExpress接続のデータベースアプリケーションを作成するのと同じであることも分かるであろう。【図8】

ところで、設定が完了したら、一度 tblDTMSCP の Active プロパティを True に変更して、データベースに正しく 接続できるか確認しておくとよい。無事 に接続確認ができたら、いったん SQLConnection1のConnectedプロパティ をFalseに戻して、接続を切断しておこう。 これだけで、DataSnap サーバプログラ

ムの作成は完了である。

完了したらプロジェクトに名前を付けて保存してからコンパイルのうえ、完成したExeファイルを実行してみよう。すると、フォームが1つ立ち上がるアプリケーションが実行されるだろう。実は、このプログラムが、DataSnapを使用したアプリケーションサーバとなっているのである。

では、このサーバプログラムを立ち上 げた状態のまま、次項からクライアント プログラムを作成していこう。

4. DataSnapクライアント プログラム作成手順

クライアントプログラムは、VCL フォームアプリケーションとして作成し よう。

新規作成後、生成された Form1 に TClientDataSet、TDataSource、TDBGrid および TDBNavigator を 貼 り付けて、各コンポーネントの紐付けを 行う。この手順も、dbExpress 接続アプリケーションで、お馴染みであろう。

● TSQLConnection コンポーネント

次に、TSQLConnection コンポーネントを貼り付け、ConnectionName プロパティに DataSanpCONNECTION を選択しよう。実は DataSnap クライアントプログラムは、dbExpress 接続プログラム同様、TSQLConneciton で接続を行うのである。Driver プロパティにはサーバ情報を指定すればよい。【図9】

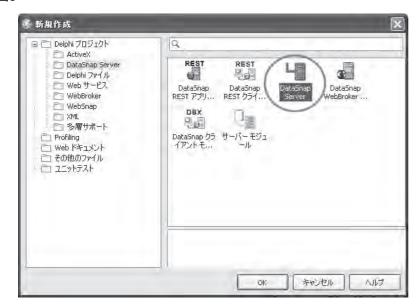


図4





具体的には、今回は同じ端末上に DataSnap サーバが稼働しているため、 HostName プロパティは localhost のままでよい。(DataSnap サーバプログラムがリモートサーバの場合は、サーバの IP アドレスを指定することとなる。) また、DataSnap サーバプログラムで指定したポート番号「211」を、Port プロパティに設定しよう。

DataSnap サーバプログラムへの接続 設定が完了した。

● TDSProviderConnection コンポーネント

続いて、TDSProviderConnection コンポーネントを貼り付けよう。これは、DataSnap サーバプログラムで作成したサーバメソッドクラスを指定するものだ。

ここでは、ServerClassName プロパティに TServerMethods1 と入力し、SQLConnectionプロパティに SQLConnection1 を選択すればよい。これで、DataSnap サーバ上のサーバメソッドをクライアントから使用できるようになる。

最後に、ClientDataSet1を選び、RemoteServerプロパティにDSProviderConnection1を選択する。すると、ProviderNameプロパティに、DataSnapサーバプログラム上で定義したdspDTMSCPが選択できるはずである。ここまでのところを一通り設定したのが図10である。【図10】

以上で、設定が完了である。

設定が完了したら、ソース 2 のようなプログラムを記述しよう。画面起動時(Form の OnCreate 時)にクライアントデータセットを開く処理と、データセット Post 後(ClientDataSet の OnAfterPost 時)にクライアントデータセットの変更内容をサーバに適用するロジックのみである。【ソース 2】

完成したら、クライアントアプリケーションを実行してみよう。クライアントプログラムには、IBM i (AS/400) に接続する処理は一切記述されていないが、データが表示されているのが分かるであろう。【図 11】

ここまで、DataSnap を使用すること で、簡単に3層アプリケーションを構築 できることをお伝えした。DataSnap の 仕組みと設定ポイントがお分かりいただ けたかと思う。

5.Windowsサービスを 使用したDataSnap サーバの作成

先ほど作成した DataSnap サーバプログラムは、VCLフォームアプリケーションであった。つまり、クライアントプログラム実行前に、あらかじめ DataSnapサーバプログラムの画面を起動しておく必要がある。しかし、実際のアプリケーションサーバでは、常にログインしたままプログラムを実行しておくというのは現実的ではない。

そこでここからは、DataSnap サーバ プログラムを「Windows サービス」と して実行できるようにしてみよう。

また、先ほどは単純に得意先マスタの 内容を表示するだけであったが、今度は クライアントアプリケーション側から絞 り込み条件として「得意先カナ」を指定 することで、対象データを検索できるよ うに拡張してみたい。

● Windows サービス

Windows サービスプログラムとは、Windows 動作中にバックグラウンドで常に実行させることが可能なプログラムのことである。これはログインしなくても実行させることができるため、DataSnap サーバプログラムに最適であろう。

Windows サービスを使用した DataSnap サーバプログラムは、前述の 「3. DataSnap サーバプログラム作成手順」の項と同様の手順でプロジェクトが 作成できる。

ウィザードの第1画面目(図4)に出る「プロジェクトの種類」で「サービスアプリケーション」を指定すればよい。ただし、作成されたプロジェクトは、前述の「VCLフォームアプリケーション」と同様だが、Unit1.pas(Form1)だけがない点に注意されたい。今回の「サービスアプリケーション」には通常画面が存在しないからだ。

●データ検索処理

また今回は、クライアントからデータ

検索処理が呼び出せるようサーバメソッドも追加しよう。

宣言部に、GetDataDTMSCPという 名前の関数を追加する。【ソース 3】

この関数の引数(AMCTRKN)に指定されたカナ名をもとに得意先マスタをカナ名で検索し、条件に合致するデータのみ抽出できるようにしたい。データの抽出には、SQL文を発行するためにTSQLQueryコンポーネントを使用しよう。ServerMethodsUnit1には図12のように設定を行う。【図12】

準備ができたら、ソース4のようなロジックを作成しよう。ここでは、サービス起動時にデータベースへ接続する処理と、引数に指定された「カナ」を条件にデータを抽出するSQL文を発行し、検索結果が1件以上存在した場合、戻り値にTrueをセットするメソッドを作成している。【ソース4】

なお、ここまで見てきた手順も、データモジュールに関数 (メソッド) を定義する手順と同様であることがお分かりいただけるであろう。

以上、ソースが完成したらプロジェクトに名前を付けて保存のうえ、コンパイルを行おう。加えて、完成したプログラム(Exe)はそのまま実行するのではなく、サービスに登録する必要がある。これは通常コマンドプロンプトで、プログラム名の後ろに「/install」を付加して実行すればよい。その後サービス管理画面で、登録されたサービスを「開始」にする。【図 13】

今回は実行しても、上述したようにプロジェクトにフォームがないため、画面が何も表示されないが、サービス実行状態が「開始」となっていれば完了である。

6.サーバメソッドを呼び 出すクライアントプロ グラムの作成

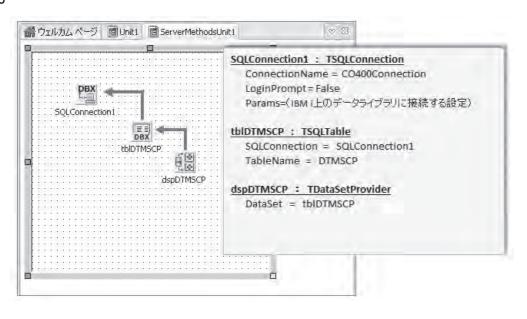
今度は、サービス登録した DataSnap サーバを使用するクライアントプログラ ムを作成していこう。

まず、検索条件となる「カナ」入 力を行う TEdit と、検索を実行す る TBitBtn を貼り付けて画面を作 成しよう。なお、TSQLConnection、 TDSProviderConnection、TClientDataSet



図7





の部分は、前述の「4. DataSnap クライアントプログラム作成手順」の項で設定した手順と同様でよい。【図 14】

● TSQLServerMethod コンポーネント 次に、TSQLServerMethod コンポーネントを貼り付けよう。これは、DataSnap サーバに定義したメソッドを呼び出すことができるクライアントコンポーネントである。具体的には、SQLConnectionプロパティにSQLConnection1を指定した後、ServerMethodNameプロパティを選ぶとメソッドの一覧が表示される。ここで、先ほど作成したGetDataDTMSCPを選択すればよい。【図15】

これで、クライアントプログラムから サーバメソッドが使用可能になるわけ だ。プログラムの記述例は、ソース5の ようになる。

メソッドの引数(今回の場合、AMCTRKN) および処理結果の戻り値 (Result) は、TSQLQuery におけるパラメータクエリー同様、ParamByName メソッドでアクセスできることが分かる。また、戻り値 (Result) は、パラメータ名の ReturnParameter にて取得可能である。なお、サーバメソッドの実行は ExecuteMethod メソッドを使用する。【ソース5】

今回作成したクライアントプログラムを実行すると、図 16 のようになる。クライアント PC で指定した条件により、DataSnap サーバが検索処理を行い、結果のデータセットを返却していることが分かる。【図 16】

7.最後に

今回は DataSnap を使用した、3 層アプリケーションの構築技法を紹介した。この技法を用いれば、例えば GUI アプリケーションと Web アプリケーションとで同じビジネスロジックを使用するような場合にも有益そうである。

図 17 は、DataSnap サーバプログラムにアクセスするクライアントプログラムを、Web アプリケーション用フレームワークである VCL for the Web を使用して作成したものである。【図 17】

Web サーバは通常、社内 LAN 環境

とは別に DMZ(非武装地帯)に配置することが多い。しかし、DataSnap を使用すれば、Web サーバとアプリケーションサーバとの間にデータベース接続用の特別なポートを開放することなくアプリケーションを公開できるため、安全性を向上させることが可能である。【図 18】

このように、DataSnapを使用した3層アプリケーションとしてシステムを構築しておくと、たとえ当初はクライアントプログラムを GUI アプリケーション 用に作成したとしても、将来 Web アプリケーション化等を検討する際に容易に応用がきくし、安全性の向上というメリット獲得にもつながる。そういったことも勘案し、ぜひ一度 DataSnap を使用した3層アプリケーションにチャレンジしてみてほしい。

M

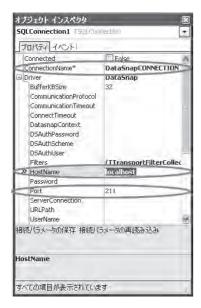
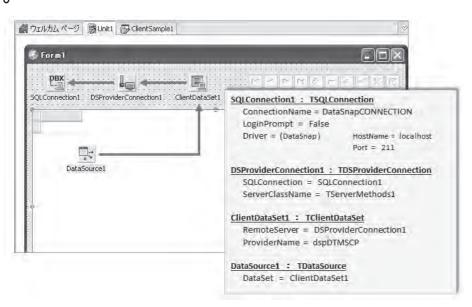


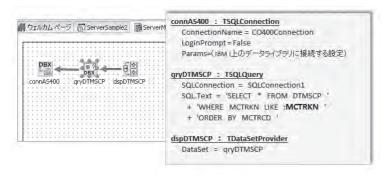
図10



ソース2

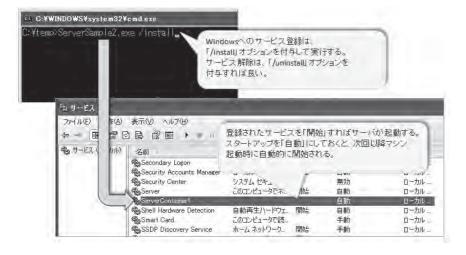


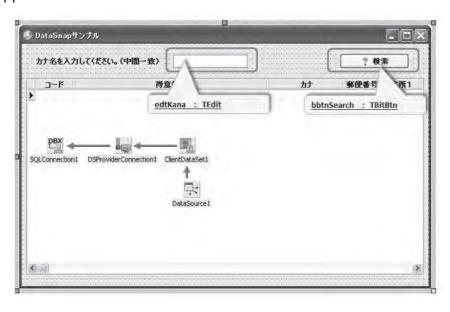
ソース3

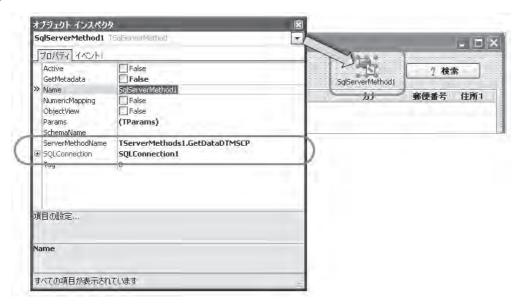


ソース4

図13

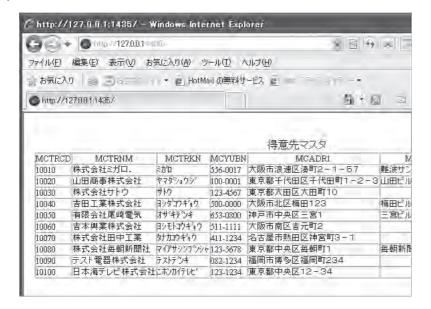


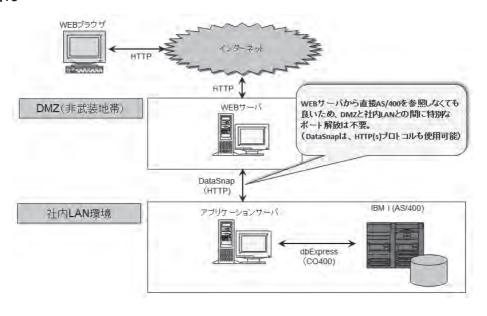




ソース5







清水 孝将 / 伊地知 聖貴

株式会社ミガロ.

システム事業部 システム1課

JC/400でポップアップウィンドウの制御&活用ノウハウ!

JC/400 の標準機能と JavaScript の 2 つのアプローチにより、 ポップアップの開発手法、活用例を紹介する。

- ●Web開発におけるポップアップウィンドウの活用
- ●JC/400標準機能による実現
- ●JavaScriptによる実現
- ●JavaScriptでの活用例
- ●補足:Ajaxを活用してHTMLを取得する
- ●まとめ



略歴 1983年10月04日生 2008年甲南大学文学部卒 2008年株式会社ミガロ. 入社 2008年04月システム事業部配属

現在の仕事内容 入社5年目でDelphi/400やJC/400 の開発業務を担当。Web に関する 知識や技術を身につけ、Web アプ リケーションのスペシャリストを目 指している。



略歴 1988年10月13日生 2011年立命館大学映像学部卒 2011年株式会社ミガロ. 入社 2011年04月システム事業部配属

現在の仕事内容 Delphi/400 や JC/400、RPG の 開発を担当。幅広いプログラム言語 を身につけ、さまざまなニーズに対 応できる SE を目指している。

1.Web開発における ポップアップウィンドウ の活用

「ポップアップウィンドウ(以下、ポップアップ)」とは、画面上のボタンクリックなどによって新たに立ち上がるウィンドウのことである。

例えば、一覧でレコードを表示する機能と、一覧から選択したレコードの詳細内容を表示する機能を、Webで実現する場合を想定してほしい。この場合、一覧画面でボタンをクリックすることで、すべてが詳細画面に切り替わってしまうと、一覧画面の他のレコード情報と比較して詳細画面を見ることができなくなってしまう。このような場合に、ポップアップを有効に活用できる。

Web でのポップアップの表示には、2 種類の方法がある。

●新規ウィンドウ

1つ目は、新規ウィンドウとして表示する方法である。新規ウィンドウの生成

は、JavaScriptのwindow.open()メソッドを用いることで簡単に実装できるため、誰でもすぐに実現が可能である。

ただし、この方法はブラウザやセキュリティソフトの機能によって、ポップアップがブロックされることが多い。その理由は、この機能が昨今、フィッシング広告としても使われており、対策としてブロック機能が用意されているからである。その結果、最近ではこの形式のポップアップは使用されることが少なくなっている。

● HTML にタグ要素追加

2つ目は、HTML上へ新たなタグ要素を追加することで、同じウィンドウ上にポップアップとして表示する方法である。この方法は、JavaScriptのコーディングを必要とするが、新規ウィンドウが立ち上がらないため、ポップアップをブロックされることがないという特徴を持っている。

JC/400 標準のポップアップ機能には、 2つ目の方法を採用している。

2.JC/400標準機能に よる実現

JC/400 では、HTMLへ新たなタグ要素を追加することで、同じウィンドウ上にポップアップが表示可能である。

実装方法も非常に簡単である。HTMLを JC/400 Designer で配布する際に、画面右上にある「新規ウィンドウを開きますか?」のチェックボックスをオンにし、その画面の表示位置(top,left)、幅(width)、高さ(height)、クラス(適用する CSS)を設定して配布するだけである。【図 1】【図 2】

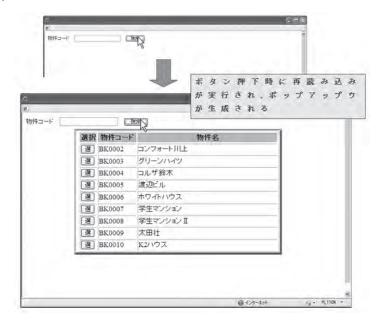
また、ポップアップとして配布することによる RPG のコーディングへの影響はなく、ポップアップであることを意識せず、通常画面と同じ手法で開発することが可能である。

3.JavaScriptによる 実現

JC/400 では、IBM i にリクエストを



図2





送り、そこで HTML を再作成することにより、ポップアップを実現している。 そのため、ポップアップを表示する際には、画面の再読み込みが必要である。

IBM i で HTML を再作成する理由 は、ポップアップに IBM i の DB 情報 を反映するからである。したがって、 DB 情報を取得しなくてもよい簡易な画 面をポップアップさせたい場合は、再読 み込みの必要がなくなる。

例えば、図3のような物件情報を照会する画面で、各物件の間取り画像を表示するようなポップアップのケースでは、必要な情報は画像ファイルの保管パスだけである。それだけの情報ならば、呼出元画面の読み込み時に取得しておけば、再度 DB 情報を取得する必要がない。そして再読み込みがなくなれば、ポップアップの表示速度が上がり、明細ごとの画像の確認がスムーズになる効果も期待できる。【図3】

このような、簡易なポップアップの表示は、JavaScript を利用することで実現可能である。【図 4】

次項から、JavaScript を用いて、再 読み込みの発生しないポップアップの表 示に必要なコーディングを解説する。 【ソース1】

3-1. タグ要素の作成

ポップアップのもととなるタグ要素を 作成する。このタグ要素を HTML 上に 追加することで、ポップアップが表示で きる。

HTMLの記述は、ソース2のように、
<BODY> タグの中にブロック要素の
<DIV> やインライン要素の
タグを記述し、さらにその中に
<TABLE> や <INPUT> タグを記述するといった階層構造になっている。【ソース2】

また、親要素への処理は、そのまま子 要素にも適用されるという特性を持って いる。親であるタグ要素を HTML に追 加すると、その子であるタグ要素も追加 される。

この親要素は、ソース1の①にある document.createElement (タグ要素名) メソッドによって、「ParentTag」という変数名で作成している。

3-2. タグ要素へ子要素を追加

作成した Parent Tag の子要素となる タグを追加する。追加する子要素をソース1の②で「Child Tag」という変数名 で作成している。

ParentTag と ChildTag は、どちらもタグ要素だが、ソース1の①と②では、作成方法が異なる。それは、①がオブジェクト型として定義されており、②が文字型として定義されている点である。

理由は、ParentTagがHTMLに記述された状態で考えると分かりやすい。

HTML 上に ParentTag を追加すると、HTML は <DIV></DIV> と記述される。この状態では、タグの中の記述がないので、画面上には何も表示されない。HTML を記述する際は、その <DIV></DIV> の中に、子となるタグ要素を記述して画面を作成していく。

これは、JavaScriptでinnerHTMLというプロパティを用いることでも可能である。innerHTMLは、タグ要素の中のHTML記述という意味のため、文字型を対象としている。つまり、<DIV></DIV> の意味を持つParentTagの中のHTML記述として、ChildTagをinnerHTMLで追加するため、ChildTagは文字型である必要があるというわけである。この処理は、ソース1の③で行っている。

また、innerHTML は文字型として親要素の中に記述するため、追加されるタグ要素の数に制限はなく、タグ要素のスタイルシートなども反映できるという特徴も持っている。

3-3. HTMLへタグ要素を追加

続いて、親要素として定義されたタグ 要素を、HTML に追加する。

タグ要素を HTML へ追加するには、 ソース 1 の ④ にある document.body. appendChild(オブジェクト型)という メソッドを用いる。これは、タグ要素を、 HTML 記述内で最上位の階層にあたる <BODY> タグの子として追加するという意味である。引数がオブジェクト型となっているのは、JavaScript 上でタグ要素を作成すると、オブジェクト型として定義されるからである。

つまり、ChildTagのような文字型変数として作成した変数は、引数として渡すことができない。この引数に、オブジェクト型として作成されたタグ要素のParentTagを指定することで、HTMLにタグ要素を追加できるようにしているのである。

以上の手順で、HTML 上へのタグ要素の追加が完了し、画面上にポップアップを表示可能になった。

なお、ソース1では、HTML にタグを追加する処理の他に、ParentTag. style.position などの記述がある。これは、HTML に追加するタグ要素のスタイルシートの設定をするための処理である。

3-4. HTMLからJavaScriptを呼び出す

ここから、ソース1の関数をHTML 上で呼び出す方法を解説する。

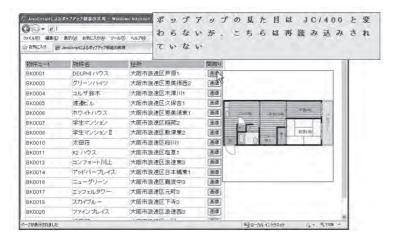
●イベントハンドラ

HTML上で JavaScript を使用するには、「イベントハンドラ」を使用する。

イベントハンドラは、画面上で発生するさまざまなアクションに対して、JavaScriptの処理を行うためのトリガーとなる命令である。これには、マウスをクリックした時に発生する on Clickや、マウスカーソルが上に載った時に発生する on Mouse Over など、さまざまな種類が用意されている。

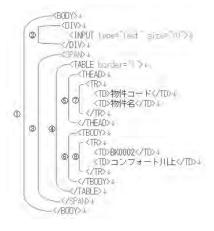
イベントハンドラは、ソース3のように HTML のタグ内に記述する。ソース3では、ボタンをクリックすると、「ボタンがクリックされました。」というメッセージダイアログが表示される。【ソース3】

また、ソース3のonClickの""で囲まれた中に、JavaScriptの処理を記述する。ここの処理には、簡単な処理以外は、関数を作成して呼び出すことが多い。理由は、イベントハンドラ内の処理はそこでしか使用できないため、別の場所で同じ処理をしたい場合に再び記述する必



ソース1

ソース2



要があり、保守性が悪くなるからである。

● PopCreate の呼び出し

ソース 4 では on Click を用いて、ソース 1 に記述された関数 Pop Create を呼び出している。 Pop Create には引数として、以下の 3 つが用意されている。 【ソース 4】

- ・iLeft (タグの X 座標)
- ・iTop (タグの Y 座標)
- ・sImgname (サムネイル画像の名称)

iLeft とiTop は、ポップアップの表示位置を指定する。ここで設定した値が、ソース1内のスタイルシートの設定に反映される。なお、表示位置は固定値で入れても問題はないが、明細の各行にボタンを設定する場合、固定値では常に同じ場所にポップアップが表示されることとなる。

今回は位置の決め方の例として、ソース5を用意した。この処理では、引数として、対象のタグ要素を渡し(this は、イベントが発生したタグ要素をオブジェクトとして指定している)、タグ要素の座標位置と画面のスクロール幅を取得することで、タグの座標位置を計算している。【ソース5】

sImgname は、表示するサムネイル画像のパスを指定する。sImgname もまた、固定値として記入してもよいが、そうすると、明細で表示した場合に、各データに対応する画像を表示することができなくなる。

対処方法として、隠しフィールドを用意し、そこへ画面読み込み時に RPG で画像名をセットすることで、各データに対応した画像表示が実現可能になる。

以上が、JavaScript でポップアップ を表示する方法となる。

4.JavaScriptでの 活用例

JavaScriptで作成するポップアップは、DB情報を取得しないため、画面の再読み込みが発生せずシームレスに表示できることが特徴である。したがって、ポップアップをHTML上の入力補助機能として活用することで、システムの運用効率を上げることも可能である。

その1例として、日付入力の補助として、カレンダーをポップアップ表示する 方法を解説する。【図5】

●カレンダーのポップアップ

ソース6では、ボタンの on Click で、Calendar という関数を呼び出している。 【ソース6】

この関数が、カレンダーのポップアップを呼び出す処理を行っている。Calendarの引数は、「タグのX座標,タグのY座標,日付を返すタグ要素」となっている。最初の2つの引数は、ポップアップを表示する位置を指定しており、ソース4と同じである。

3つ目の引数が、ポップアップしたカレンダーで選んだ日付を表示するタグ要素 を 指 定 し て い る。document. getElementById とは、HTML から指定した ID を持つタグ要素を探し、それをオブジェクト型として取得することができるメソッドである。

「Calendar」関数の中では、カレンダーの表示を行っている。ソース1で変更が必要な記述は②の ChildTag だけであり、ここにカレンダーを表示するHTMLを記述すればよい。

なお、カレンダーを表示する HTML の作成については、月の移動など複雑なコーディングが必要なこともあり、ここでは説明を割愛させていただく。カレンダー作成のノウハウは、インターネットや書籍に豊富に存在しており、それらを参考にするのもよいと思われる。

JavaScript によるポップアップ表示の手法は、さまざまな場面で活用できる。 例として挙げたカレンダー以外にも、システム開発で役立ててほしい。

補足:Ajaxを活用して HTMLを取得する

ソース1では、追加するタグ要素を JavaScript ソース内にそのまま記述し た。今回は簡易な画面だったが、ポップ アップ表示する HTML ソースの量が多 い場合、ソースの量が膨大になり、保守 性が悪くなる。

そこで、この補足の項では Ajax を活用し、外部記述された HTML をポップアップ表示する方法を解説する。

Ajax とは「Asynchronous JavaScript

+ XML」の略であり、一般的に、サーバと XML 形式のデータのやり取りを行うものである。ただし、必ずしもデータが XML 形式である必要はなく、TXT、CSV、HTML 形式等のデータも取得可能となっている。また、Ajax も JavaScript なので、新たに別の言語を使う必要がなく扱いやすい。

今回は、この Ajax の機能を用いて、 外部記述された HTML ファイルを取得 する方法を解説する。

補足1. Ajaxオブジェクトの作成

Ajax でサーバと通信を行うためには、 まず専用のオブジェクトを作成する必要 がある。【ソース 7】

ソース7の①では、専用のオブジェクトである Active XO bject を作成している。サーバとの通信は、このオブジェクトを介して行う。ただし、このオブジェクトは、Microsoft が提供しているブラウザ Internet Explorer でのみサポートされているオブジェクトである。

それ以外のブラウザで使用する場合には、XMLHttpRequestというオブジェクトが用意されている。ソース7の②では、IE 以外のブラウザの場合に、そのオブジェクトを使用して作成するようになっている。

補足2. サーバとの通信

オブジェクトの作成が完了したら、次 は実際にサーバと通信を行う。

ソース8の①では、データの取得にopenメソッドを使用している。この引数に、「メソッド,URL,同期方式,ユーザー名,パスワード」を指定し(ユーザー名・パスワードは必要な場合のみ)、次の行でsendメソッドを実行することでサーバに対してリクエストが実行される。【ソース8】

●同期・非同期

上記の引数の中に、「同期方式」というものが表示されている。これは Ajax の特徴ともいえるものであり、以下に説明する。

Ajax の通信方式には、「同期」「非同期」 の2種類が用意されている。同期型の通 信の場合、次の処理はサーバからの応答

ソース3 ⟨INPUT type="button" onClick="alert('ボタンがクリックされました。')"> ソース4 <TD><INPUT type="button" onClick="PopCreate(getElementPosition(this), right, getElementPosition(this)_top, this.parentNode.parentNode.childNodes[4].inperHTML);" value="画像"></TD>4 <TD style="display: none;">IMAGO1.jpg</TD>+ ソース5 ソース 5 function getElementPosition (obj)[+ var html = dosument.documentElement; + //画面内座標を取得→ var rect = obj.getBoundingClientRect();+ var rectleft = rect.left - html.clientLeft;+ var recttop = rect.top - html.clientTop;+ //スクロール幅を取得+ var scrollLeft = document.body.scrollLeft;+ var scrollTop = document.body.scrollTop; + //ベージ内での絶対座標を算出。 var left = rectleft + scrollLeft;+ var right = rectleft + scrollLeft+ abj.offsetWidth+ var top = recttop + scrollTop;4 var bottom = recttop + scrollTop + obj.offsetHeight+ return [left:left, right:right, top:top, bottom:bottom];+ 図5 4 4 x D3 ファイルゼ 編集型 表示似 お気に入り仏 ツールロ ヘルフセ ・ お気に入り 参加ンダー たカレンダー — Windows Internet Explorer (A)(C) = (B) 🕶 😘 🗶 👂 Search G ファイル(E) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H) ⇒ お気に入り 適力レンダー ☆・ 🗟 - 🖾 👵・ページ®・セーフティ⑤・ツール◎・ 💁・

が返ってきてからとなる。一方、非同期型の通信は、サーバからの応答を待たず、そのまま次の処理が実行される。非同期型処理のメリットは、容量の重いデータを取り扱う場合にも、サーバからの応答が完了する前に次の処理を実行できるという点が挙げられる。

補足3. 非同期型通信について

同期型と違い、非同期型通信では、サーバからの応答状況を手動で管理する必要がある。

サーバからの応答状況を知るには、onreadystatechangeというプロパティを使用する。このプロパティで設定した処理は、サーバからの応答状況が変化するたびに実行される。また、サーバからの応答状況を知るには、readyStateとstatusというプロパティが用意されており、このプロパティ値をonreadystatechangeによって実行される処理で調べることで、データの取得が完了したタイミングで次の処理を実行できる。これらの処理を行っているのが、ソース8の②である。【ソース8】

サーバより返ったデータは、response Textで文字型として取得できる。これをソース1の②で記述しているタグ要素の代わりに、ChildTagへ代入すれば、取得した外部 HTML ファイルを、ポップアップとして表示することができる。

5. まとめ

ポップアップは、さまざまな場面において活用することが可能である。JC/400の標準機能を用いて、マスタ検索などを実現する方法と、JavaScriptを用いて、DBとの連携を必要としない簡易な画面を実現する方法の2種類を紹介させていただいた。

この2つを場面によって上手に使い分けることで、Webシステムで実現できる機能の幅は大きく広がっていくだろう。Web画面と聞くとブラウザによる制約が大きいというイメージを抱かれる方が多いかもしれないが、JavaScriptをうまく活用することができれば、思った以上に柔軟にさまざまなことをWebで実現可能にできる。

昨今は、HTML5の登場により、

Webでの可能性がさらに広がっている。本稿を第一歩として、成長を続ける Webの世界へ飛び込んでいただければ 嬉しい。

M

ソース6

ソース7

```
目的: 非同期通信用オブジェクトの作成。
引数: なし+
戻値: なし+
function createXMLHttpRequest() {+
// IEの場合。ActiveXObjectを使用+
  if (window.ActiveXObject) (+
   Try (*
// MSXML2.XMLHTTPが使える場合は、使用する(M)crosoft.XMLHTTPよりも高速)**
return XMLhttpObject = new ActiveXObject("Mayer7-XMLHTTP");4
    catch(e) {+
      try { 4
// MSXML2.XMLRTTPが使えない場合は、Microsoft.XMLHTTPを使用する。
                                                                               1
      return XWLhltpObject = new ActiveXObject("Microsoft-XWLHTTPを使用する。
return XWLhltpObject = new ActiveXObject("Microsoft-XWLHTTP");4
)*
      catch(e) (+
     return null; +
     IE以外の場合、XMLHttpRequestを使用+
      if (window.XMLHttpRequest) (+
  return XMLhttpObject = new XMLHttpRequest();+
                                                    2
 return mult;+
  else I-
```

ソース8

Migaro. Technical Report 既刊号バックナンバー

Migaro. Technical Report 各号の電子版は、ミガロ. のホームページでご覧いただけます。 http://www.migaro.co.jp/

No.1 2008 年秋

お客様受賞論文

●最優秀賞

直感的に理解できるシステムを目指して一情報の"見える化" の取り組み

石井 裕昭様/豊鋼材工業株式会社

●ゴールド賞

運用部間にサプライズをもたらした Delphi/400

春木 治様/株式会社ロゴスコーポレーション

●シルバー賞

JACi400 使用による Web アプリケーション開発工数削減中富 俊典様/日本梱包運輸倉庫株式会社

Delphi/400 を利用した Web 受注システム

飯田 豊様/東洋佐々木ガラス株式会社

●優秀賞

Delphi/400 による販売管理システム (FAINS) について 藤田 建作様/株式会社船井総合研究所

技研化成の新基幹システム再構築

藤田 健治様/技研化成株式会社

SE 論文

はじめての Delphi/400 プログラミング

畑中 侑/システム事業部 システム 2 課

Delphi/400 と Excel との連携

中嶋 祥子/RAD 事業部 技術支援課

連携で広がる Delphi/400 活用術

尾崎 浩司/システム事業部 システム 2 課

フォーム継承による効率向上開発手法

吉原 泰介/ RAD 事業部 技術支援課

API を利用した出力待ち行列情報の取得方法

鶴巣 博行/ RAD 事業部 技術支援課

Delphi テクニカルエッセンス Q&A 集

吉原 泰介/ RAD 事業部 技術支援課

JACi400 を使って RPG で Web 画面を制御する方法

松尾 悦郎/システム事業部 システム 2 課

あなたはプラインドタッチができますか?

福井 和彦/システム事業部 システム 1 課

No.2 2009 年秋

お客様受賞論文

●最優秀賞

JACi400 で 既存 Web サービスの内製化を実現

佐々木 仁志様/株式会社ジャストオートリーシング

●ゴールド賞

.NET 環境での Delphi/400 の活用

福田 祐之様/林兼コンピューター株式会社

●シルバー賞

5250 で動作する「中古車 在庫照会プログラム」の GUI 化 佐久間 雄様/株式会社ケーユー

●優秀賞

Delphi による 輸入システム「MISYS」の再構築

秦 榮禧様/株式会社モトックス

Delphi/400 による 物流システムの再構築

仲井 学様/西川リビング株式会社

Delphi/400 で開発し 3 台のオフコンを 1 台の IBM i へ統合 島根 英行様/シルフ

SE論文

JACi400 環境でマッシュアップ!

岩田 真和/ RAD 事業部 技術支援課

Delphi/400 を利用したはじめての Web 開発

福岡 浩行/システム事業部 システム 2 課

Delphi/400 を使用した Web サービスアプリケーション

尾崎 浩司/システム事業部 システム 3 課

Delphi/400 によるネイティブ資産の応用活用

吉原 泰介/ RAD 事業部 技術支援課 顧客サポート

RPG でパフォーマンスを制御

松尾 悦郎/システム事業部 システム 1 課

MKS Integrity を利用したシステム開発

宮坂 優大 田村 洋一郎/システム事業部 システム 1 課

No.3 2010年秋

お客様受賞論文

●最優秀賞

建物のクレーム情報管理システム「アフターサービス DB」 について

大橋 良之様/東レ建設株式会社

●ゴールド賞

Delphi/400 で「写真管理ソフト」と「スプールファイル の PDF 化ソフト」を自社開発

寒河江 幸喜様/日綜産業株式会社

●シルバー賞

Delphi/400 で鉄鋼受発注業務を統一し 鉄鋼 EDI も実現 柿本 直樹様/合鐵産業株式会社

●優秀賞

Delphi/400 で EIS (Executive Information System) の高速化

小島 栄一様/西川計測株式会社

イントラでの PHP-Delphi-RPG 連携

仲井 学様/西川リビング株式会社

Delphi/400 を使った取引先管理システム

大崎 貴昭様/森定興商株式会社

SE 論文

Delphi/400 ローカルキャッシュ活用術

中嶋 祥子/ RAD 事業部 技術支援課

Delphi/400 帳票開発ノウハウ公開

尾崎 浩司/システム事業部 システム 3課

Delphi/400 でドラッグ&ドロップを制御

辻林 涼子/システム事業部 システム 2 課

Delphi/400 のモジュールバージョン管理手法

前田 和寛/システム事業部 システム 2 課

Delphi/400 Web からの PDF 出力

福井 和彦・清水 孝将/システム事業部システム 3 課・システム 2 課

Delphi/400 で Flash 動画の実装

吉原 泰介/ RAD 事業部 技術支援課 顧客サポート

No.4 2011 年秋

お客様受賞論文

●最優秀賞

全社の経費処理業務を効率化した「e 総務システム」 鈴木 英明様/阪和興業株式会社

●ゴールド賞

「Web 進捗管理システム」でリアルタイム性を実現 堀内 一弘様/エスケーロジ株式会社

●シルバー賞

「営業奨励金申請書」をたった2日間で開発

簑島 宏明様/株式会社ケーユーホールディングス

液体輸送における「配車支援システム」の構築

桂 哲様/ライオン流通サービス株式会社

SE 論文

グラフ活用リファレンス

中嶋 祥子/RAD事業部 技術支援課

Web サービスを利用して機能 UP!

福井 和彦・畑中 侑/システム事業部 システム 2 課

OpenOffice 実践活用

吉原 泰介/ RAD 事業部 技術支援課 顧客サポート

VCL for the Web 活用 TIPS 紹介

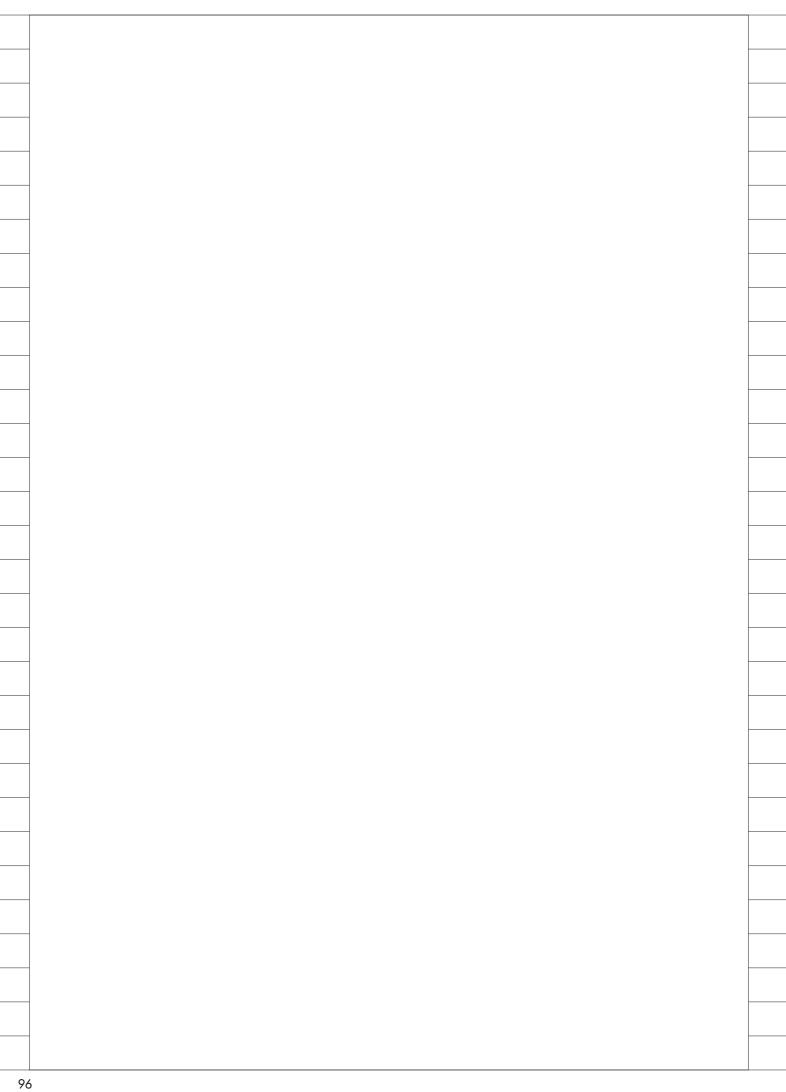
尾崎 浩司/システム事業部 プロジェクト推進室

JC/400 で JavaScript 活用

清水 孝将/システム事業部 システム 1 課

jQuery 連携で機能拡張

國元 祐二/ RAD 事業部 技術支援課 顧客サポート



MIGARO. TECHNICAL REPORT

Migaro.Technical Report

No.5 2012 年秋

ミガロ.テクニカルレポート

[創刊 5 周年記念号]

2012年11月1日初版発行

◆発行

株式会社ミガロ.

〒 556-0017

大阪府大阪市浪速区湊町 2-1-57 難波サンケイビル 13F

TEL: 06(6631)8601 FAX: 06(6631)8603

http://www.migaro.co.jp/

◆発行人

上甲 將隆

◆編集協力

アイマガジン株式会社

◆デザインフォーマット

近江デザイン事務所

©Migaro.Technical Report2012

本誌コンテンツの無断転載を禁じます

本誌に記載されている会社名、製品名、サービスなどは一般に各社の商標または

登録商標です。本誌では、TM、®マークは明記していません。



MIGARO. TECHNICAL REPORT

ミガロ.テクニカルレポート











株式会社三ガロ

http://www.migaro.co.jp/

本社

〒556-0017

大阪市浪速区湊町2-1-57 難波サンケイビル 13F

TEL:06(6631)8601

FAX:06(6631)8603

東京営業所

〒106-0041

東京都港区麻布台1-4-3

エグゼクティブタワー麻布台 11F

TEL:03(5573)8601 FAX:03(5573)8602





