

【セッションNo. 1】

Delphi/400開発ノウハウお教えします

## Delphi/400による『帳票』開発ノウハウ公開 &システム開発事例紹介

株式会社ミガロ.  
システム事業部 システム3課  
尾崎 浩司

## 【アジェンダ】

### 『帳票』開発ノウハウ公開

- 1) 「RaveReports」による帳票開発手法
- 2) PDF出力技法
- 3) 将来を見越した帳票開発

### システム開発事例紹介

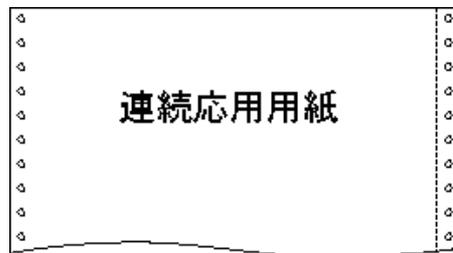
- 4) AS/400スプールとの連携
- 5) ドラッグ & ドロップによる簡便なGUI操作

## Delphi/400による 『帳票』開発ノウハウ公開

## ■ Delphi/400による帳票開発

### ● 『帳票』開発どうしていますか？

- Delphi/400ユーザー ⇒ IBM I (AS/400) ユーザー
  - IBM I(AS/400) を使用
    - ⇒ プリントファイルを使用



### ● 『QuickReport』を使用

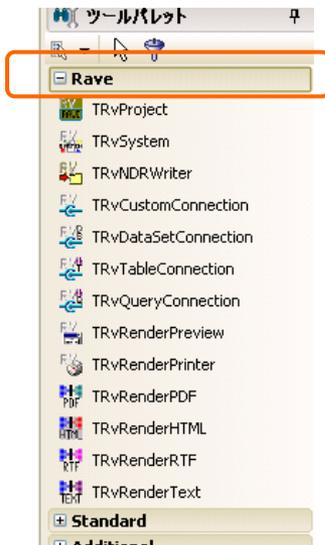
- ⇒ V5 ~ V7の頃は、主流
- V2005以降はバンドル無し

※ V2006、V2007の場合、エンバカデロよりを追加  
モジュールとしてダウンロード可能

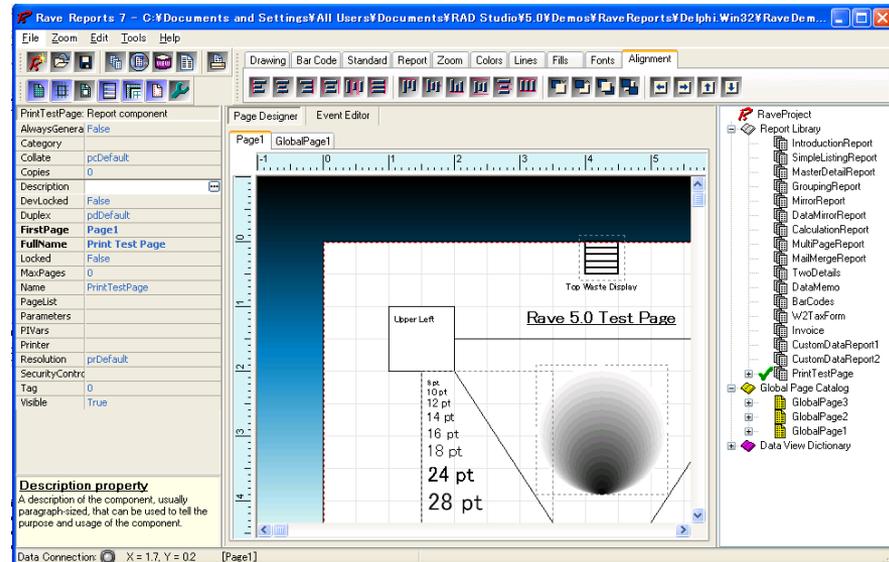


## ■Delphi/400による帳票開発

- 現在のDelphi/400標準レポートツールは『RaveReports』
  - Ver.7 よりDelphiにバンドル
  - Delphiコンポーネント群 + レポートデザイナー で構成



+



## ■ RaveReports特徴

### ● RaveReportsとQuickReportとの違い

|            | Rave Reports   | Quick Report   |
|------------|--|--|
| 帳票設計       | <p>デザイナーにて作成</p> <p>Delphiツールメニューより呼出可能</p>          | <p>Delphiフォームとして作成</p> <p>フォーム上にQRReportコンポーネントを配置</p> |
| 帳票ファイル     | <p>レポートファイルを使用</p> <p>RaveReportsファイル(拡張子rav)が生成</p> | <p>Exeファイルにレポートが含有</p> <p>実行プログラムに含まれる</p>             |
| データセット     | <p>利用可能</p> <p>データセットをデザイナー上で関連付けることが可能</p>          | <p>利用可能</p> <p>データセットコンポーネントと関連付けされる印字コンポーネントが使用可能</p> |
| プログラムからの制御 | <p>可能</p> <p>関連付けたレポートをDelphi言語から制御可能</p>            | <p>可能</p> <p>帳票フォームを生成し、印刷メソッドを実行</p>                  |
| その他        | <p>30,000件データの一覧リスト</p> <p>プレビュー 約10秒</p>            | <p>30,000件データの一覧リスト</p> <p>プレビュー 約15秒</p>              |

## ■ RaveReports 特徴

- Delphiソースコードで制御ができる。
- DataSetを帳票に指定できる。
- レポートファイルが分離されたため、帳票レイアウト変更がDelphiプログラムに影響しない。
- 複数のレポートをひとつのレポートファイルで管理が可能。
- QuickReportに比べ出力速度が速い。(調査比150%)

---

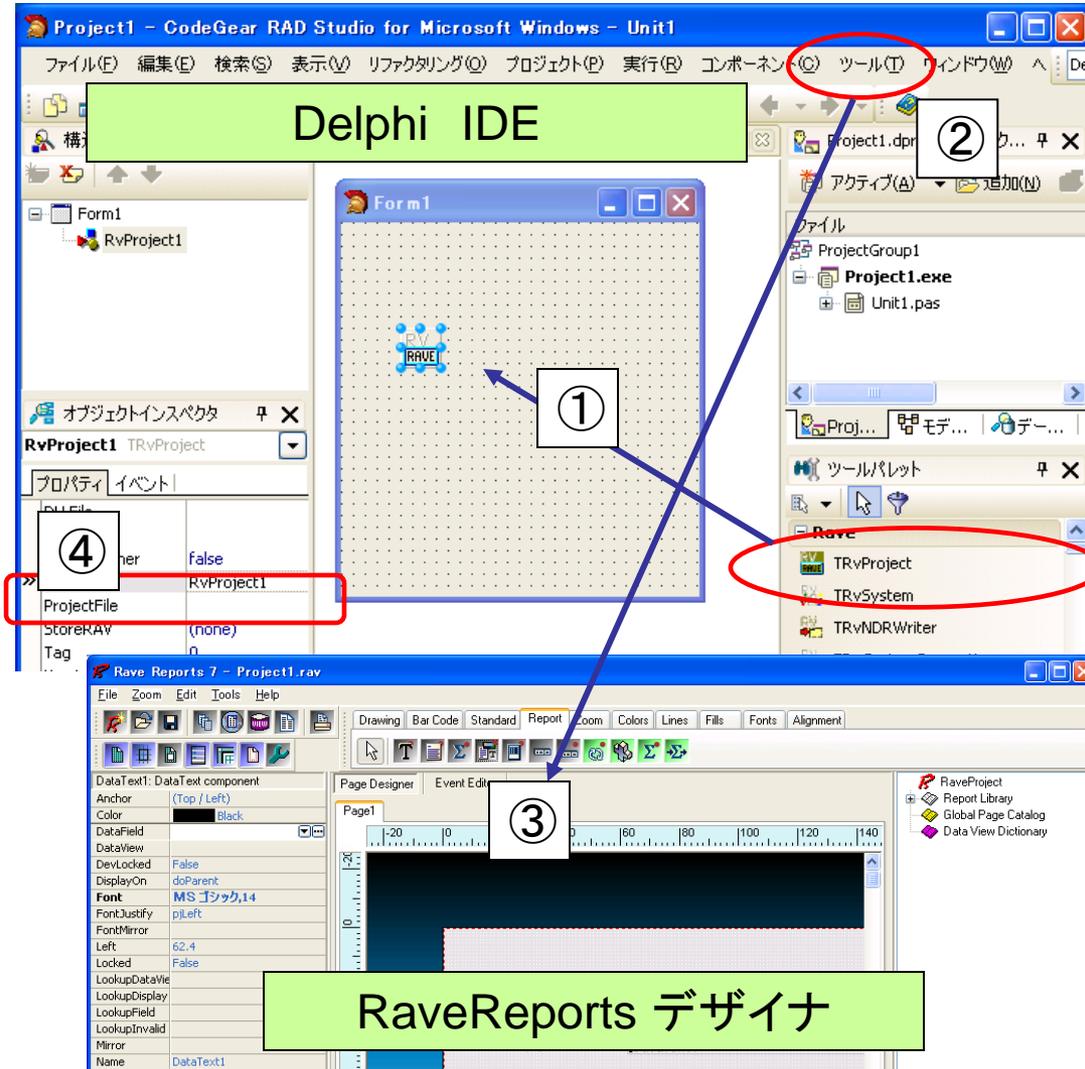
### ● 日本語環境での問題

- 通常の帳票出力は、日本語使用可能
- PDF出力機能は、日本語使用不可
- レポートデザイナー画面は、英語環境のみ

## ■ RaveReports開発手順

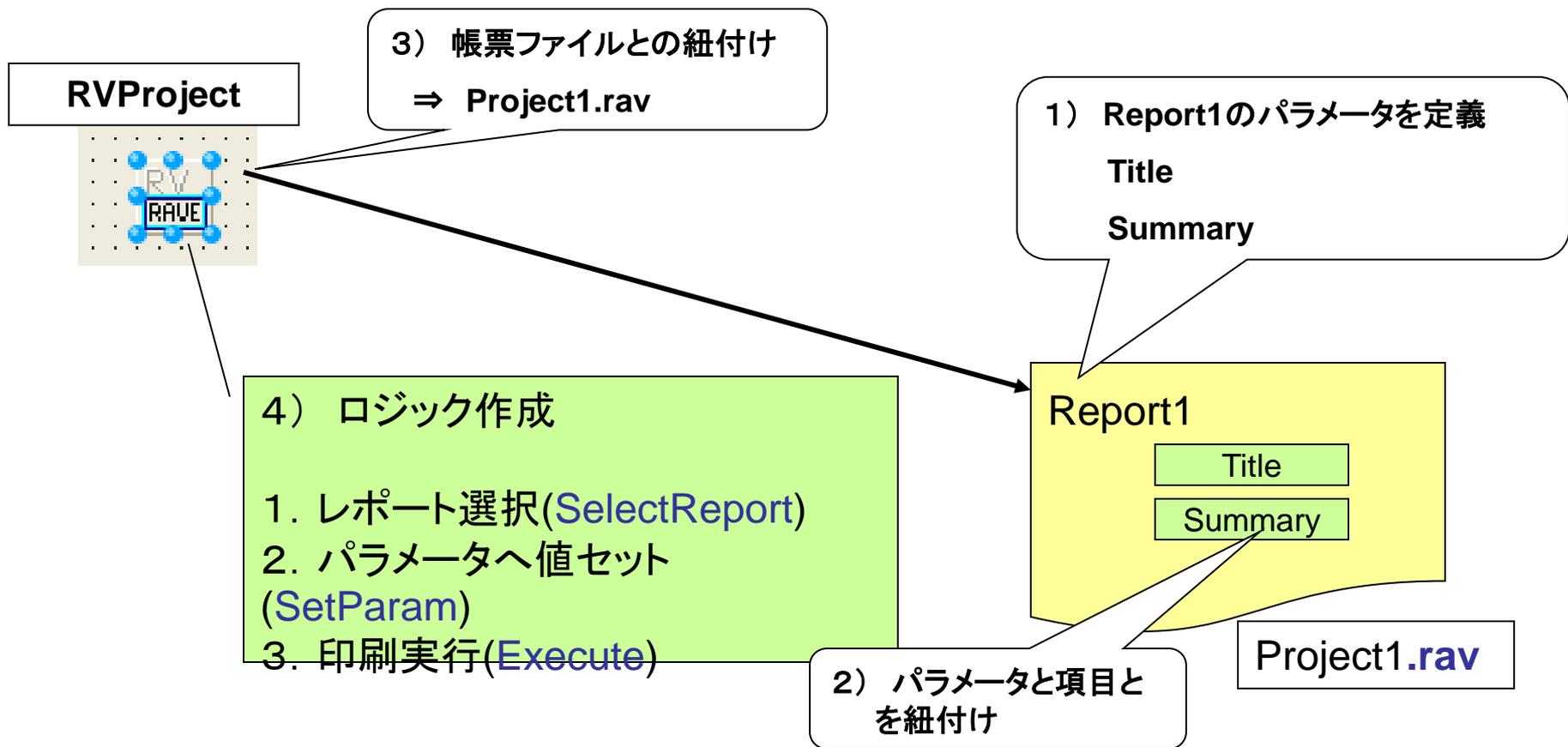
### ● RaveReports作成の流れ

- ① TRVProjectコンポーネントの貼り付け
- ② RaveReportsデザイナを起動  
→ [ツール]⇒  
[RaveReportsDesigner]
- ③ レポートの設計、保存  
→ レポートファイル生成
- ④ RvReportコンポーネントにレポートファイル関連付け  
→ ProjectFileプロパティ
- ⑤ 出力ロジック追加  
→ SelectReportプロパティ  
→ Executeメソッド



## ■ RaveReports 最初の一步

### ● 概念図



## ■ RaveReports 最初の一步

### ● 画面設計

#### ● Standardページ

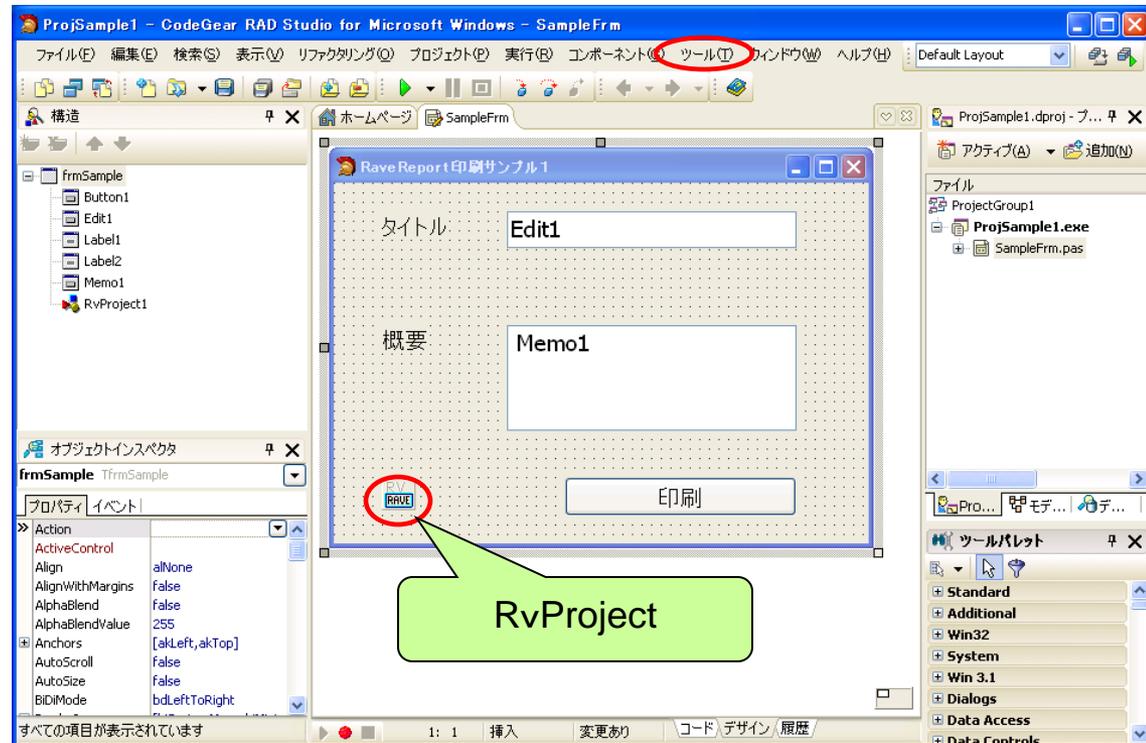
- Labelコンポーネント
- Editコンポーネント
- Memoコンポーネント
- Buttonコンポーネント

#### ● Raveページ

- RVProjectコンポーネント

### ● デザイナの起動

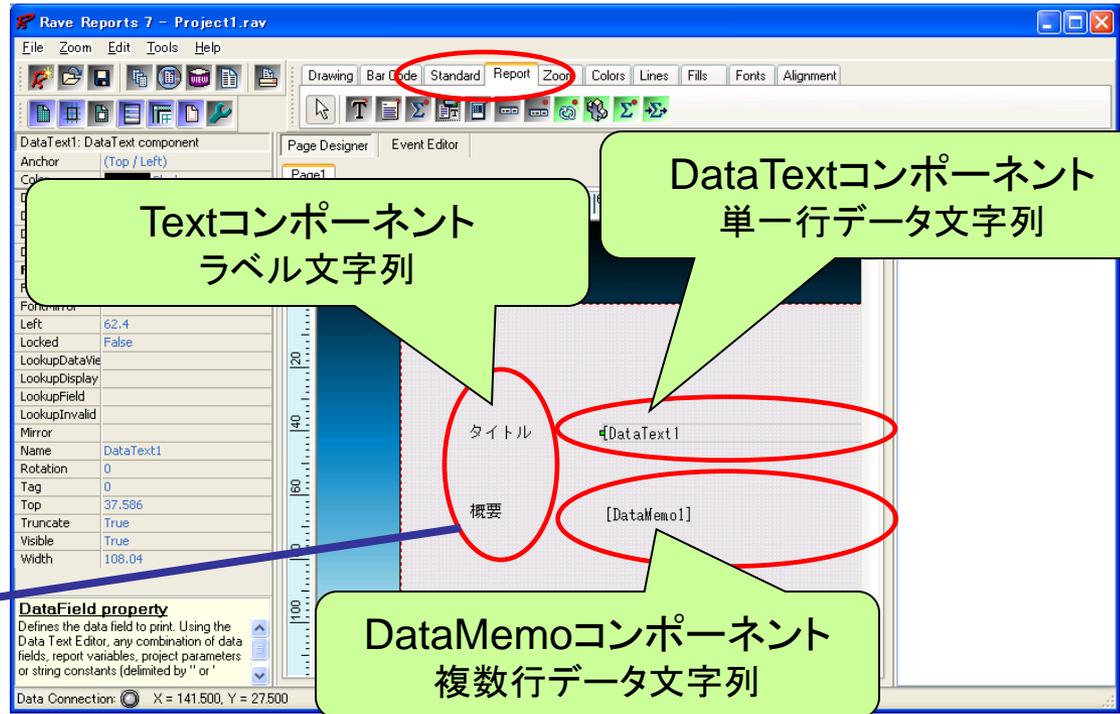
- [ツール] → [RaveReportsDesigner]



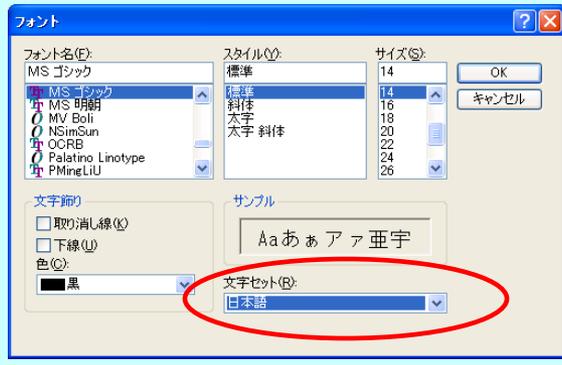
## ■ RaveReports 最初の一步

### ● 帳票設計

- Standardページ
  - Textコンポーネント
- Reportページ
  - DataTextコンポーネント
  - DataMemoコンポーネント



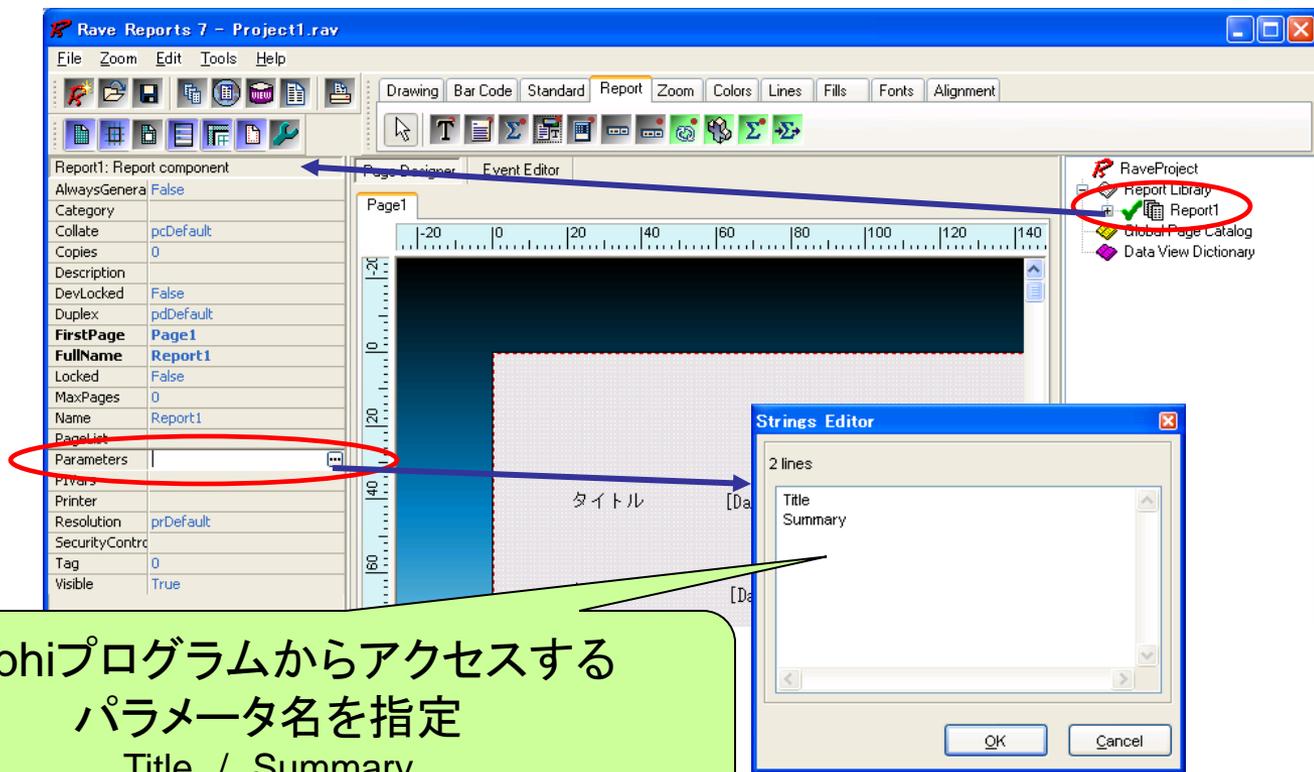
※日本語フォント時文字セットを「日本語」に指定



## ■ RaveReports 最初の一歩

### ● 帳票パラメータ設定

- Report1 選択
- Parameters プロパティ



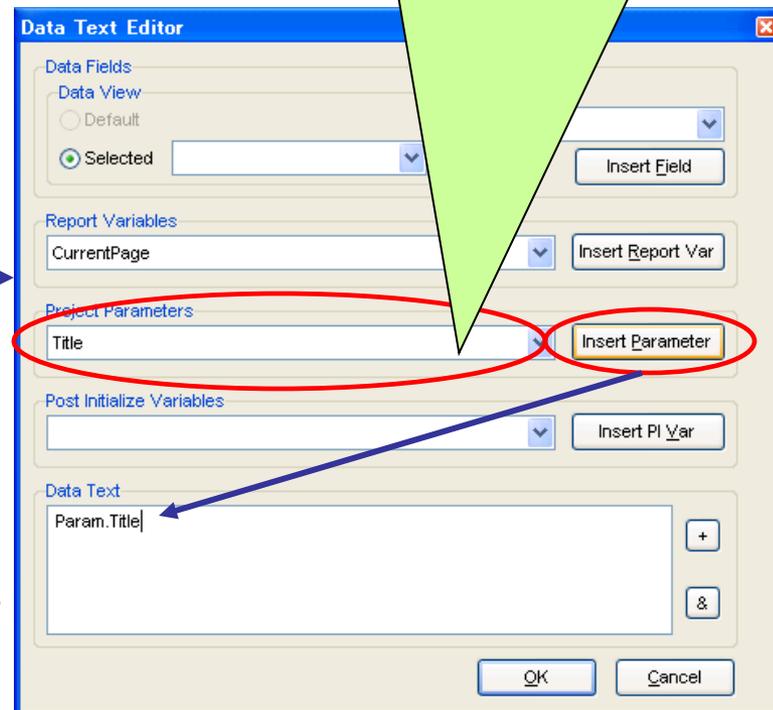
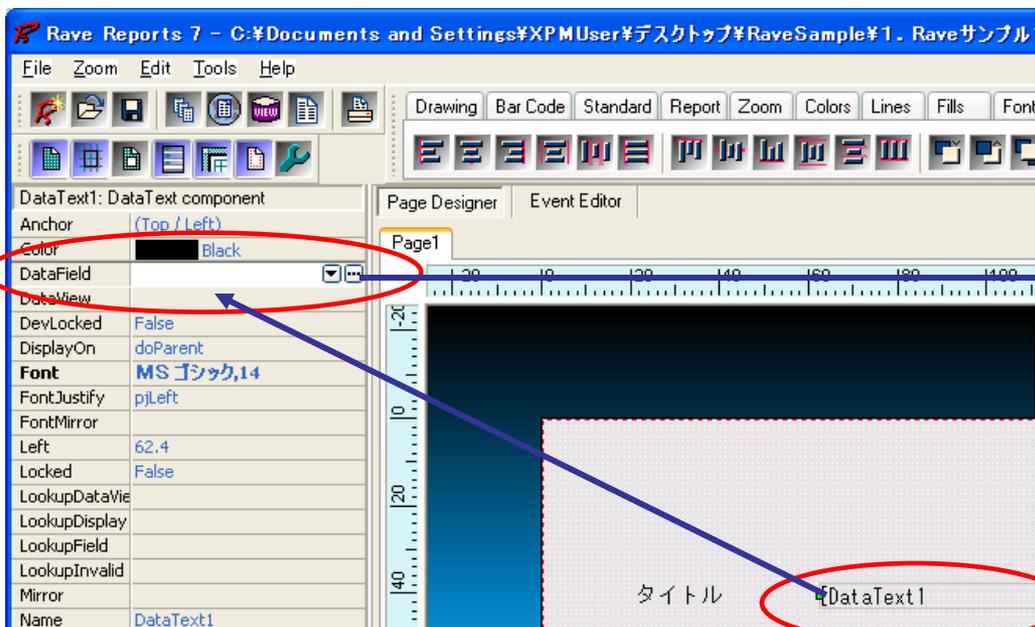
Delphiプログラムからアクセスする  
パラメータ名を指定  
Title / Summary

## ■ RaveReports 最初の一歩

### ● 帳票パラメータ設定

- DataTextコンポーネント選択
- DataFieldプロパティ

1. リストよりパラメータを選択
2. 「Insert Parameter」ボタン押下



## ■ RaveReports 最初の一步

### ● Delphi側設定

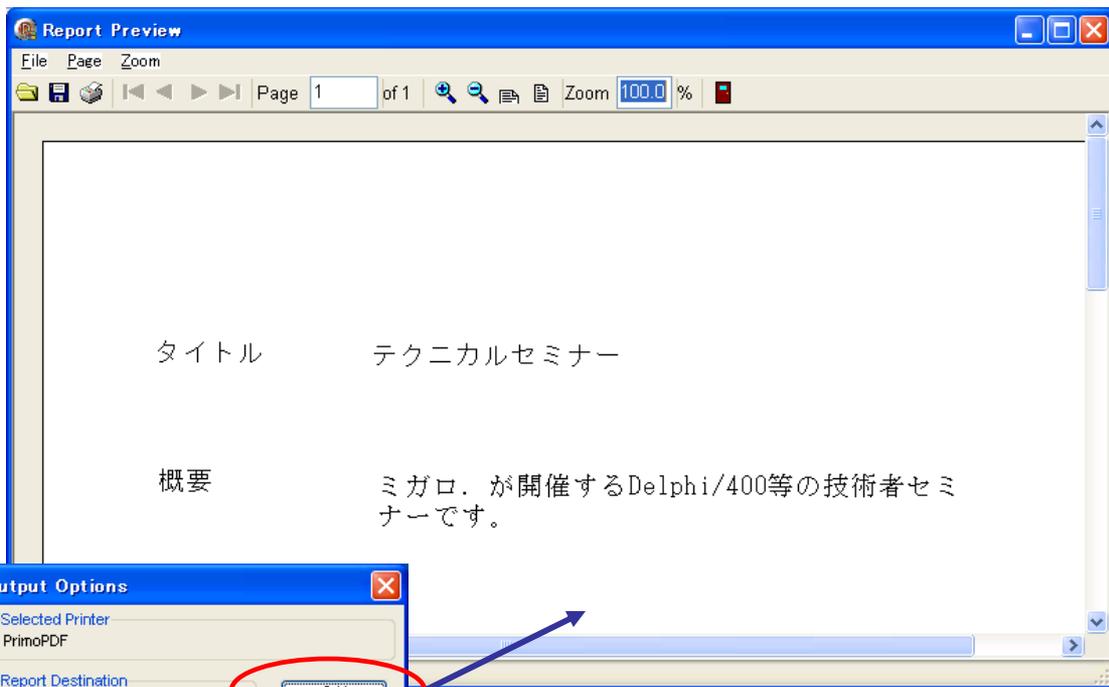
- RvProjectコンポーネント
  - ProjectFileプロパティ
- ロジック記述
  - SelectReportプロパティ
  - SetParamプロパティ
  - Executeメソッド



```
procedure TfrmSample.Button1Click(Sender: TObject);
begin
  with RvProject1 do
  begin
    // Raveプロジェクトを開く
    Open;
  try
    // レポートの選択
    SelectReport('Report1', True);
    // 値のセット
    SetParam('Title', Edit1.Text);
    SetParam('Summary', Memo1.Text);
    // 実行
    Execute;
  finally
    // Raveプロジェクトを閉じる
    Close;
  end;
end;
end;
```

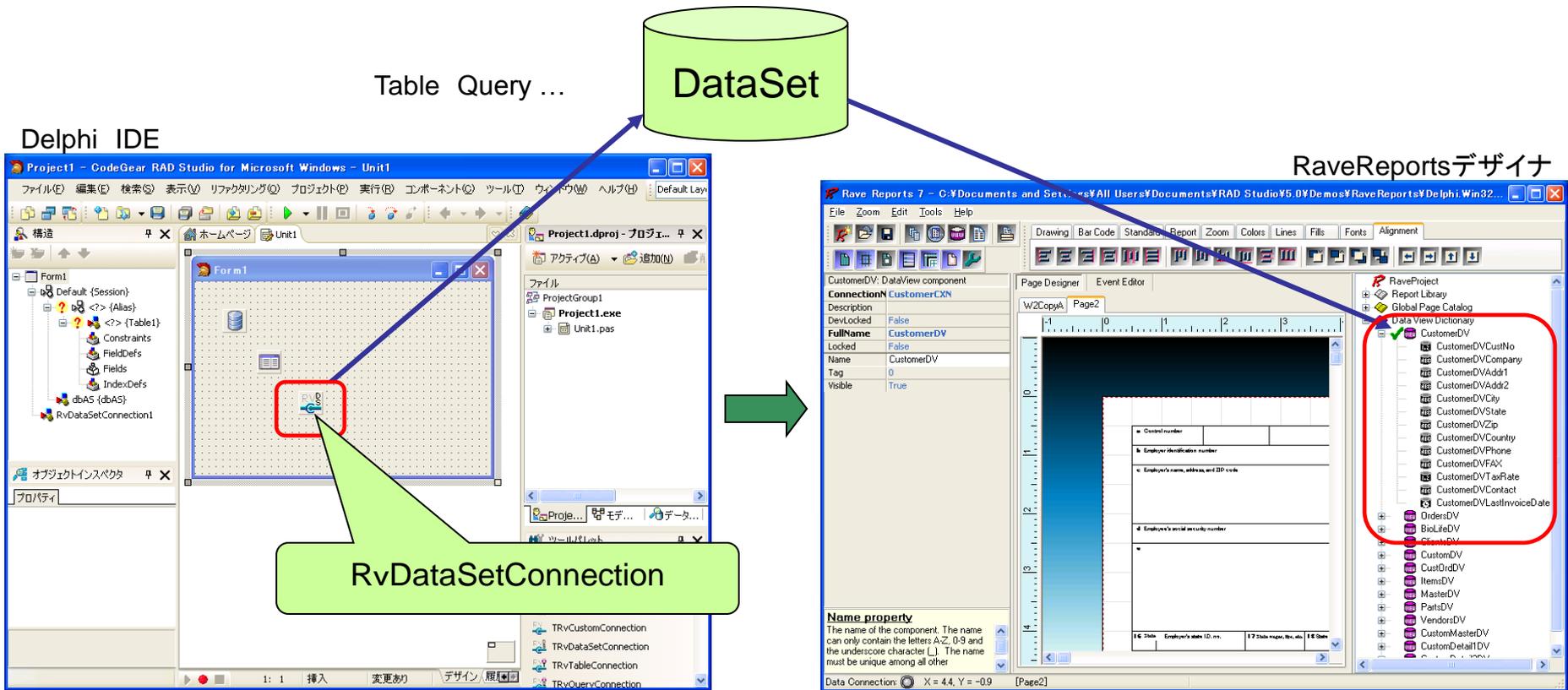
## ■ RaveReports 最初の一步

### ● 実行



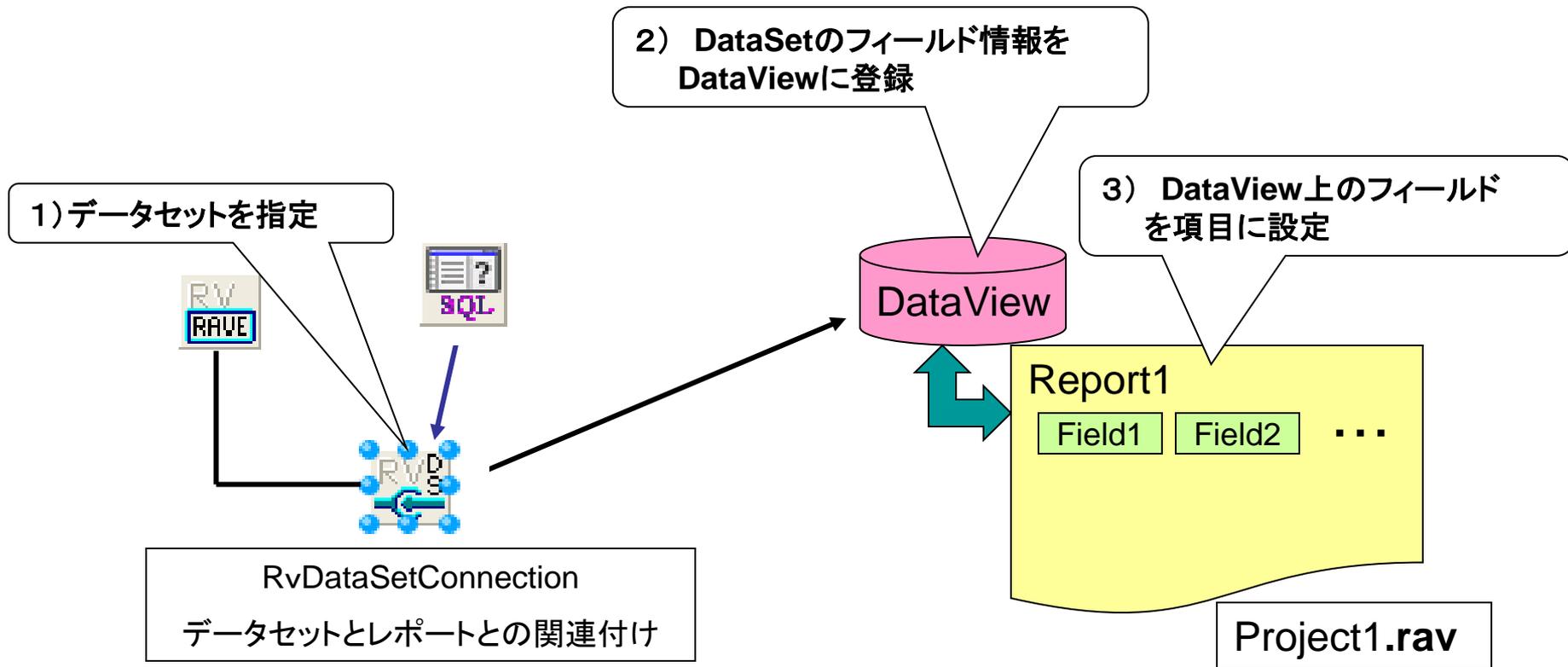
## ■ RaveReports DataSetを使用した帳票

- DataSetと連動した帳票出力が可能



## ■ RaveReports DataSet帳票

### ● 概念図



## ■ RaveReports DataSet帳票

### ● 画面設計

- Raveページ
- RvDataSetConnectionコンポーネント

RaveReportサンプル2

受注NO 1000001 表示

| 行NO | 商品CD   | 商品名                | 数量 | 単価        |
|-----|--------|--------------------|----|-----------|
| ▶ 1 | MN0001 | DELPHI/400 DEVELOP | 1  | 1,000,000 |
| 2   | MN2312 | マウス                | 20 | 2,000     |
| 3   | MN3210 | デスクトップPC           | 20 | 49,800    |
| 4   | MN3211 | 17型液晶ディスプレイ        | 20 | 24,800    |

DB Grid

RvDataSetConnection

印刷

オブジェクトインスペクタ

RvDataSetConnection1 TRvDataSetConn

DataSet Query1

DisableDataSource true

FieldAliasList (TDataSet)

LocalFilter true

Name RvDataSetConnection1

RestoreDataSet true

RuntimeVisibility rtDefault

Tag 0

Version 7.5.2

出力対象データセットを指定

すべての項目が表示されています

## ■ RaveReports DataSet帳票

### ● 帳票設計

- New Data Objectメニュー  
→ RvDataSetConnection1を指定

The image shows a sequence of three screenshots from the Rave Reports 7 software interface, illustrating the steps to configure a data set report.

- Top Screenshot:** The 'New Data Object' menu is open, and 'RvDataSetConnection1' is selected. A green callout box points to the 'Data View Dictionary' in the background, which lists several data views like 'DataView1JUJUNO'.
- Bottom Left Screenshot:** The 'Data Connections' dialog box is shown. 'Direct Data View' is selected in the list. A green callout box says 'Direct Data Viewを選択' (Select Direct Data View).
- Bottom Right Screenshot:** The 'Active Data Connections' dialog box is shown. 'RvDataSetConnection1 (DT)' is selected in the list. A green callout box says 'フォーム上の RvDataSetConnection1を選択' (Select RvDataSetConnection1 on the form). The 'Finish' button is also highlighted with a red circle.

Additional annotations include a green callout box at the top right stating 'データセット項目が Data View Dictionary に登録' (Data set items are registered in the Data View Dictionary).

## ■ RaveReports DataSet帳票

### ● 帳票設計

#### ● Report Wizardメニュー

→ Simple Tableウィザードを指定

The image illustrates the steps for creating a report in Rave Reports 7. It shows the main application window with the 'Report Wizards' menu open, highlighting the 'Simple Table' option. Below this, three sequential dialog boxes are shown, each with a callout box explaining a step:

- Dialog 1:** 'Select The Data View you wish to use for this report'. A red circle highlights 'DataView1' in the list. Callout: 対象となるデータセットを選択 (Select the target data set).
- Dialog 2:** 'Select The Fields you wish to use for this report'. A list of fields is shown with checkboxes. 'JUJUNO' is unchecked, while 'JUJUNO', 'JUJUNO', 'JUJUNO', 'JUJUNO', 'JUJUNO', and 'JUJUNO' are checked. Callout: 対象となるフィールドを指定 (Specify the target fields).
- Dialog 3:** 'Arrange Fields In The Order You Wish Them Printed'. A list of fields is shown in a specific order: JUJUNO, JUJUNO, JUJUNO, JUJUNO, JUJUNO, JUJUNO. Callout: 出力順に項目並び替え (Reorder items by output order).

## ■ RaveReports DataSet帳票

### ● 帳票設計

帳票タイトルを指定

日本語フォント指定後、「Generate」ボタン押下

タイトル、メイン、ボディ部のフォントを指定

レイアウト調整

レイアウト調整

## ■ RaveReports DataSet帳票

### ● Delphi側設定

### ● 実行

```
procedure TfrmSample.btnPrintClick(Sender: TObject);  
begin  
  with RvProject1 do  
  begin  
    Open;  
    try  
      SelectReport('Report1', True);  
      Execute;  
    finally  
      Close;  
    end;  
  end;  
end;  
end; // 次の1行でもOK  
RvProject1.ExecuteReport('Report1');
```

RaveReportサンプル2

受注NO 1000001

| 行NO | 商品CD   | 商品名                | 数量 | 単価        |
|-----|--------|--------------------|----|-----------|
| 1   | MN0001 | DELPHI/400 DEVELOP | 1  | 1,000,000 |
| 2   | MN2312 | マウス                | 20 | 2,000     |
| 3   | MN3210 | デスクトップPC           | 20 | 49,800    |
| 4   | MN3211 | 17型液晶ディスプレイ        | 20 | 24,800    |

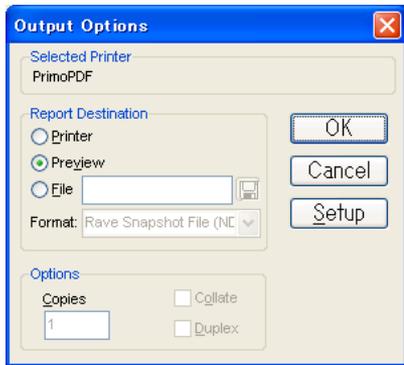
DataSetのDisplayFormatが帳票にも適用

受注商品明細

| 行NO | 商品CD   | 商品名                | 数量 | 単価        |
|-----|--------|--------------------|----|-----------|
| 1   | MN0001 | DELPHI/400 DEVELOP | 1  | 1,000,000 |
| 2   | MN2312 | マウス                | 20 | 2,000     |
| 3   | MN3210 | デスクトップPC           | 20 | 49,800    |
| 4   | MN3211 | 17型液晶ディスプレイ        | 20 | 24,800    |

## ■ RaveReports 応用テクニック①

- 帳票指示画面を使用せずに出力
  - 指示画面が英語表示のため、非表示に変更する
  - RvSystemコンポーネントを使用
    - 帳票に関する設定情報を保持



帳票指示画面  
(英語表記)

1) 指示画面表示設定  
ssAllowSetup

2) 帳票設定の関連付け

デフォルト出力先  
rdPreview  
rdPrinter  
ソースにて設定可能

## ■ RaveReports 応用テクニック①

### ● 帳票指示画面を使用せずに出力

rgSelect : TRadioGroup



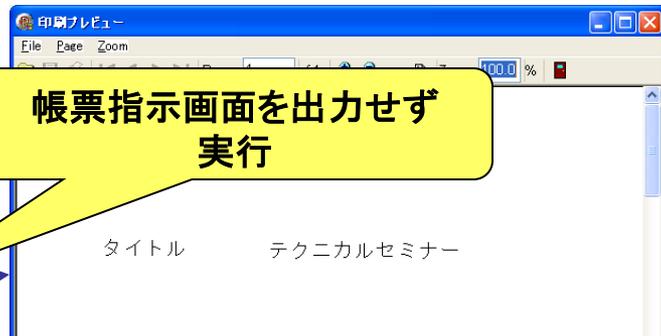
```
procedure TfrmSample.Button1Click(Sender: TObject);
begin
  // 帳票プレビュー選択
  if rgSelect.ItemIndex = 0 then
    RvSystem1.DefaultDest := rdPreview // プレビュー
  else
    RvSystem1.DefaultDest := rdPrinter; // 印刷
end;
```

3) 出力先をDefaultDestにて指定

実行  
(プレビュー表示)



帳票指示画面を出力せず  
実行



## ■ RaveReports 応用テクニック②

### ● 計算項目の使用

1) Report1のParameters  
プロパティ

パラメータ: Amount を定義



2) CalcOpコンポーネント  
を明細欄に貼り付け

CalcOp1: CalcOp component

DestParam Amount

DestPIVar

DevLocked False

DisplayForm #,0

DisplayType dtNumericFormat

Locked False

Name CalcOp1

Operator coMul

ResultFunction cfNone

Src1CalcVar

Src1DataFiel JUPRSU

Src1DataVie DataView1

Src1Function cfNone

Src1Value 0

Src2CalcVar

Src2DataFiel JUPRTN

Src2DataVie DataView1

Src2Function cfNone

Src2Value 0

Tag 0

Visible True

3) 計算式指定

DestParam(出力項目名)

Operator(演算子)

Src1...、Src2...(項目)

DisplayFormat(書式)

商品明細

数量 単価

## ■ RaveReports 応用テクニック②

### ● 計算項目の使用

4) DataTextコンポーネントを明細欄に貼り付け

5) パラメータのAmountを指定

実行(プレビュー表示)

| 受注商品明細 |           |           |
|--------|-----------|-----------|
| 数量     | 単価        | 金額        |
| 1      | 1,000,000 | 1,000,000 |
| 20     | 2,000     | 40,000    |
| 20     | 49,800    | 996,000   |
| 20     | 24,800    | 496,000   |

数量 × 単価が金額欄として出力

## ■ RaveReports 応用テクニック③

### ● 画像の挿入

2) Bitmapコンポーネント  
出力ファイルをFileLinkで指定可能

3) Bitmap1のOnBeforePrintイベント  
RaveProjectオブジェクトの  
GetParamプロパティより  
PictureFileStr値を取得

1) Report1のParametersプロパティ  
パラメータ: PuctureFileStr を定義

```
Bitmap1.FileLink :=  
RaveProject.GetParam('PictureFileStr');
```

```
function Bitmap1_OnBeforePrint(Self: TRaveBitmap);  
begin  
    Bitmap1.FileLink := RaveProject.GetParam('PictureFileStr');  
end OnBeforePrint;
```

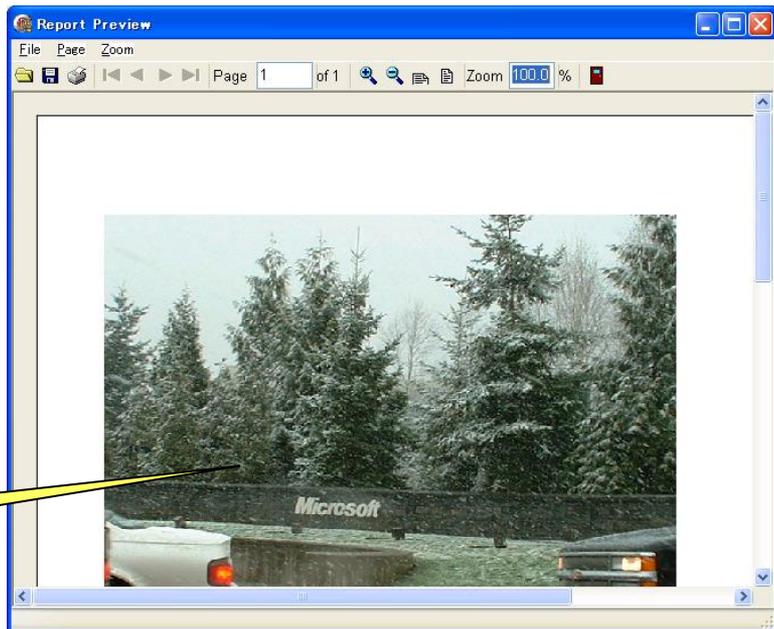
## ■ RaveReports 応用テクニック③

### ● 画像の挿入

```
procedure TfrmSample.Button1Click(Sender: TObject);
begin
  with RvProject1 do
  begin
    //レポート選択
    SelectReport('Report1', True);
    // 画像ファイル名をパラメータにセット
    SetParam('PictureFileStr', Edit1.Text);
    // 印刷実行
    Execute;
  end;
end;
```

4) 画面で指定したビットマップファイルを  
パラメータに指定

実行(プレビュー表示)



指定された画像が帳票出力

## ■PDF出力

### ● PDFファイルを生成する手法

#### ● PDFプリンタドライバを使用

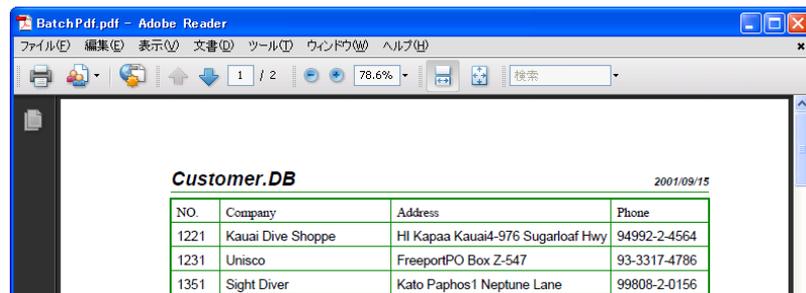
- RaveReportsやQuickReportから印刷する際に、PDF出力を選択

- AdobeAcrobat
- PrimoPDF(無償)
- PDFCreator(無償) オープンソース  
<http://sourceforge.jp/projects/pdfcreator/>



#### ● PDF作成コンポーネントを使用

- RaveReportsよりPDF出力可能だが、日本語が使用不可
- PDF専用レポートコンポーネントである「PowerPDF」を使用することで対応可能
  - Ver2009 の場合 PowerPDF0.9.1日本語版  
[http://mam.dnsalias.net/download/delphi\\_powerpdf.html](http://mam.dnsalias.net/download/delphi_powerpdf.html)
  - ~Ver2007の場合 PowerPDF0.9  
[http://www.est.hi-ho.ne.jp/takeshi\\_kanno/powerpdf/](http://www.est.hi-ho.ne.jp/takeshi_kanno/powerpdf/)



## ■PDF出力手法の比較

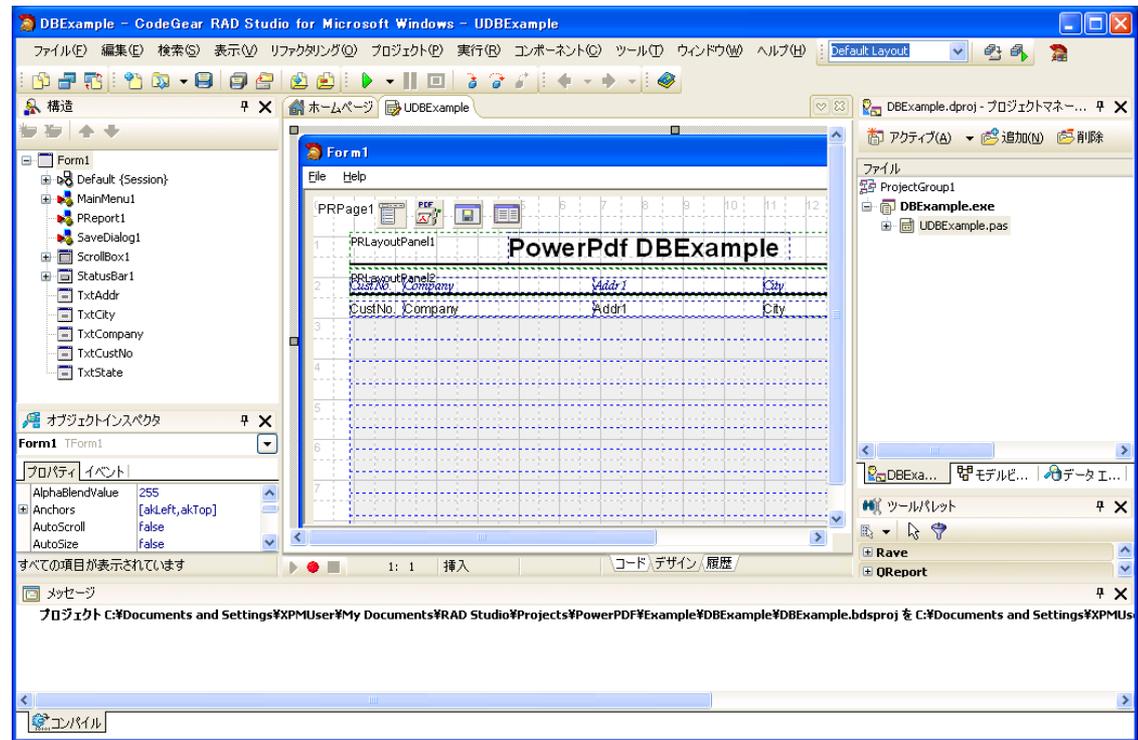
### ● 専用コンポーネントとプリンタドライバとの違い

|            | 専用コンポーネント (PowerPDF)   | プリンタドライバ (PDFCreator)   |
|------------|--|---|
| 開発手法       | PowerPDFコンポーネント使用<br>PDF用の追加プログラムを作成   | 既存の帳票プログラムを使用<br>帳票は、紙出力のものと同じものを使用   |
| 環境         | AdobeReaderのみが必要<br>閲覧用ソフトのみで使用可能  | 出力用プログラムが別途必要<br>クライアント端末に事前に設定が必要  |
| プログラムからの制御 | 可能<br>出力ファイル名、出力先等Delphiプログラムにより指定可能   | 不可<br>出力先は、都度ユーザーが指定。<br>(ドライバ設定により出力先固定化は可能)   |
|            | <ul style="list-style-type: none"> <li>・システムから明示的にPDF作成処理が作成できる</li> <li>・専用帳票作成工数が別途必要</li> </ul> | <ul style="list-style-type: none"> <li>・ユーザー指定によるPDF作成処理となる。</li> <li>・既存帳票はそのまま使用可能</li> </ul> |

## ■PowerPDF 開発手順

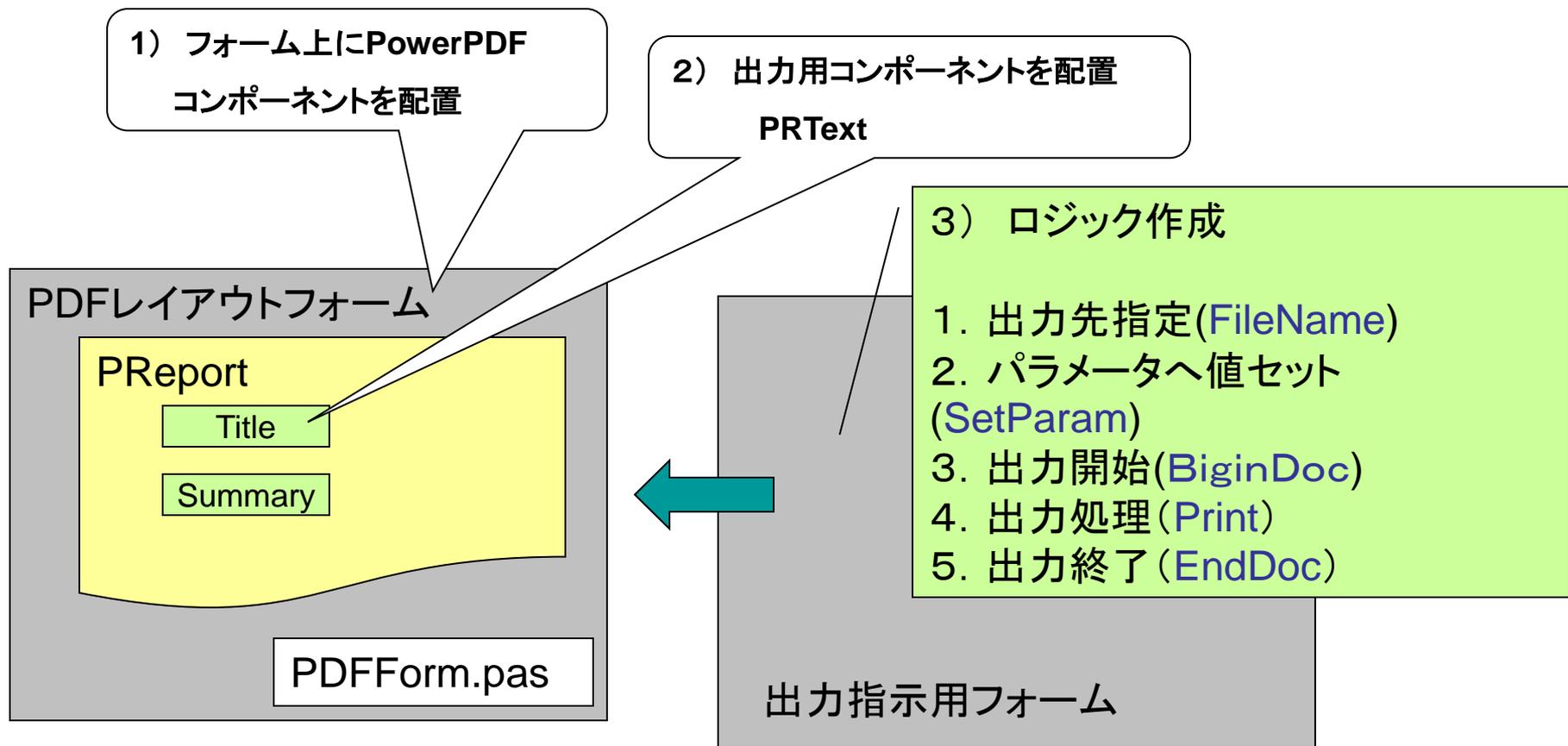
### ● PowerPDF作成の流れ

1. 帳票出力フォームにPowerPDFコンポーネントの貼り付け  
→ PReportコンポーネント
2. 帳票設計  
→ PRPage、PRTText等
3. 出力ロジック追加  
→ BeginDocメソッド  
→ FileNameプロパティ  
→ Printメソッド  
→ EndDocメソッド



## ■PowerPDF 開発手順

### ● 概念図



## ■PowerPDF 開発手順

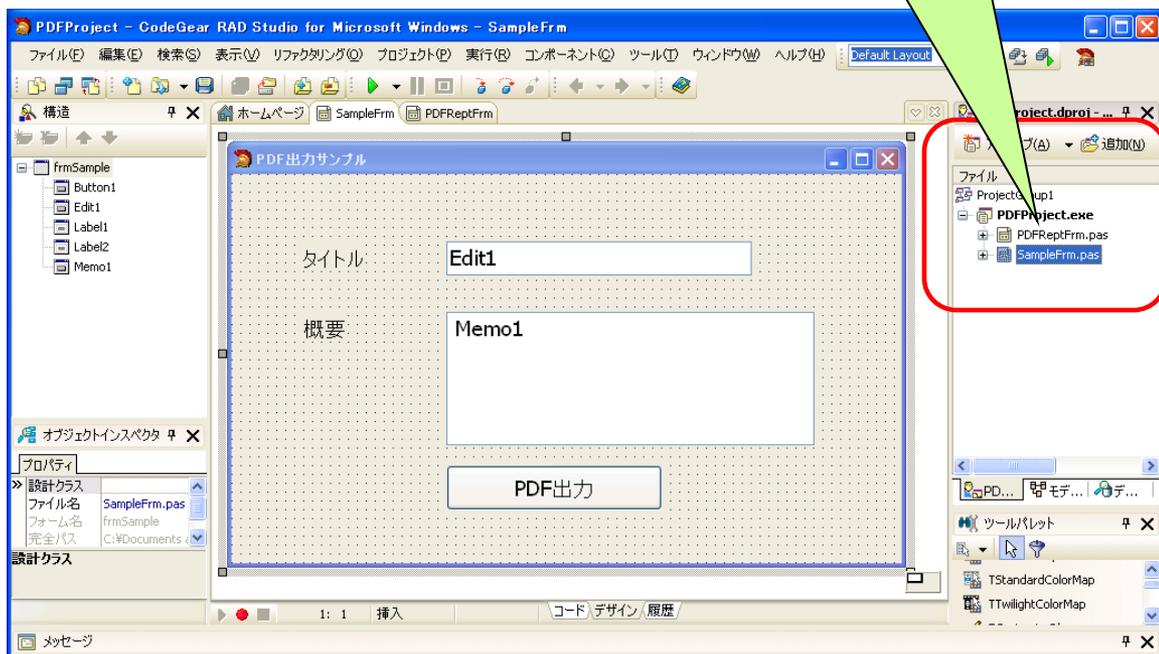
### ● 画面設計

- プロジェクトに帳票用フォーム追加
  - PDFReptFrm.pas

### ● Standardページ

- Labelコンポーネント
- Editコンポーネント
- Memoコンポーネント
- Buttonコンポーネント

帳票レイアウトフォーム  
PDFReptFrm.pas



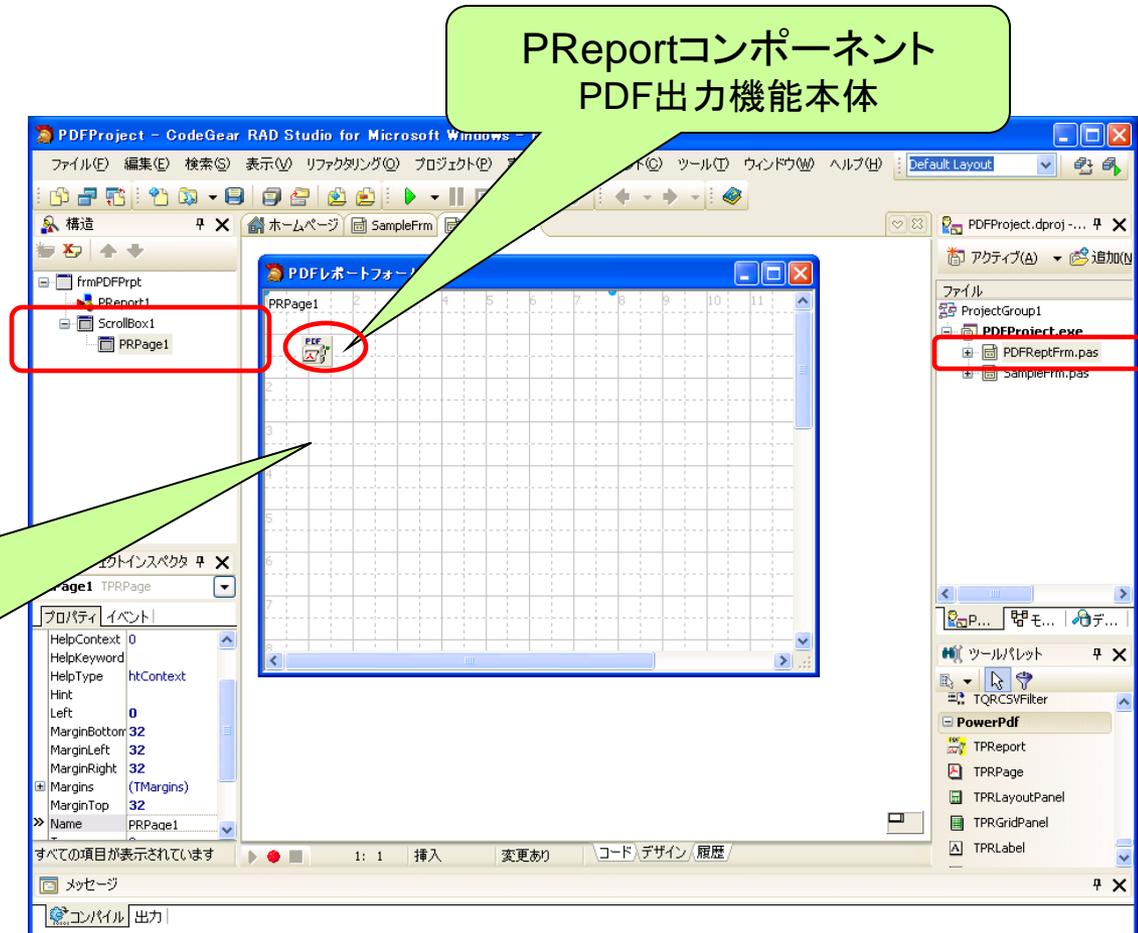
## ■PowerPDF 開発手順

### ● 帳票設計

- Additionalページ
  - Scrollboxコンポーネント
- PowerPDFページ
  - PReportコンポーネント
  - PRPageコンポーネント

PRPageコンポーネント  
PDF1ページを構成

ScrollBox (Align := alClient)を  
親コントロールに配置する



## PowerPDF 開発手順

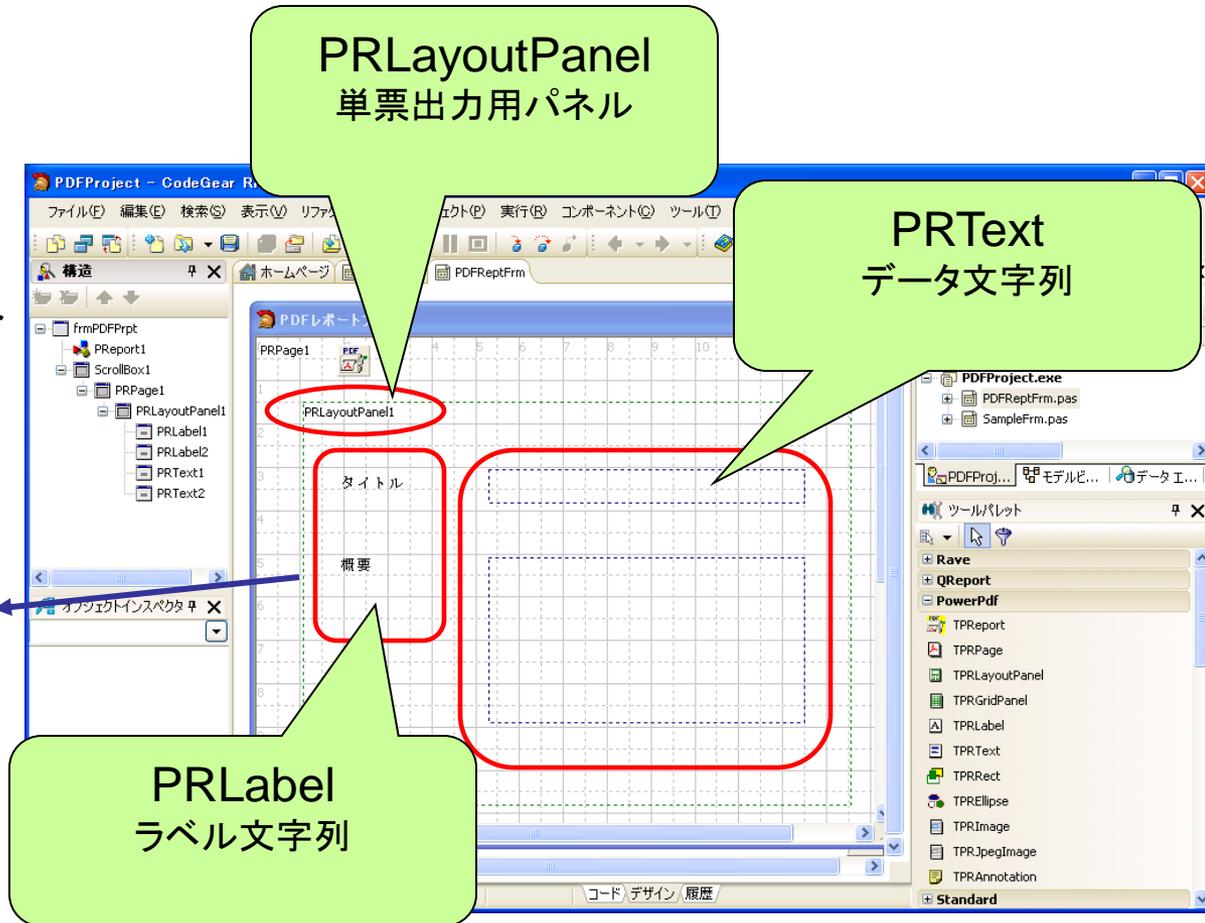
### ● 帳票設計

- PowerPDFページ
  - PRLayoutPanelコンポーネント
  - PRLabelコンポーネント
  - PRTTextコンポーネント

日本語フォントは、FontNameプロパティで指定

fnGothic、fnMincyo、  
fnPGothic、fnPMincyo

※～V2007用のPowerPDF0.9の場合、  
PReport.pasにある{\$DEFINE  
USE\_JPFONTS}を有効にしてください。



## ■PowerPDF 開発手順

### ● ソースコード

#### ● 出力指示画面プログラム

```
uses PDFReptFrm; // PDFフォーム

{$R *.dfm}

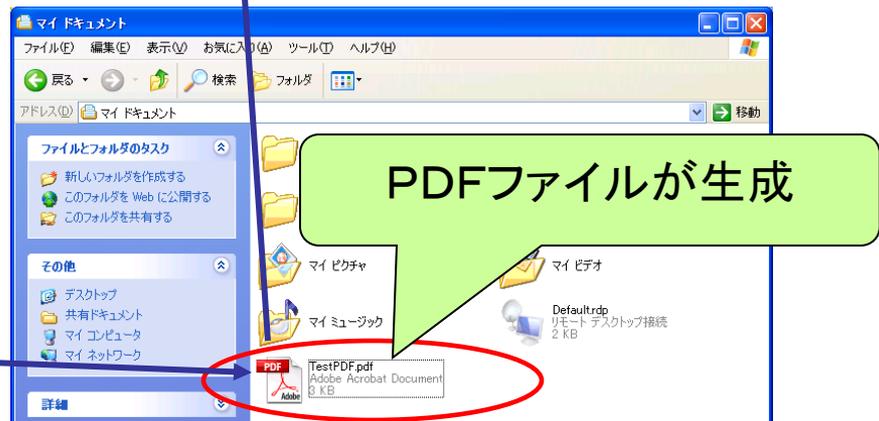
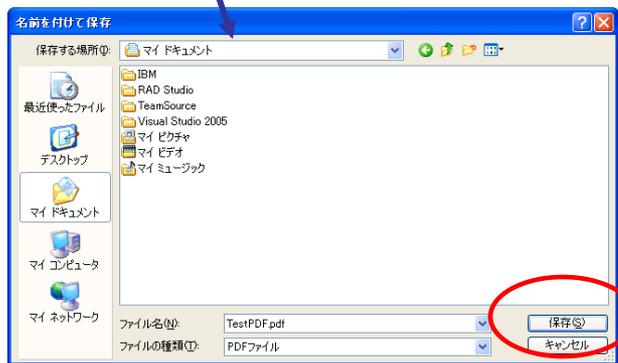
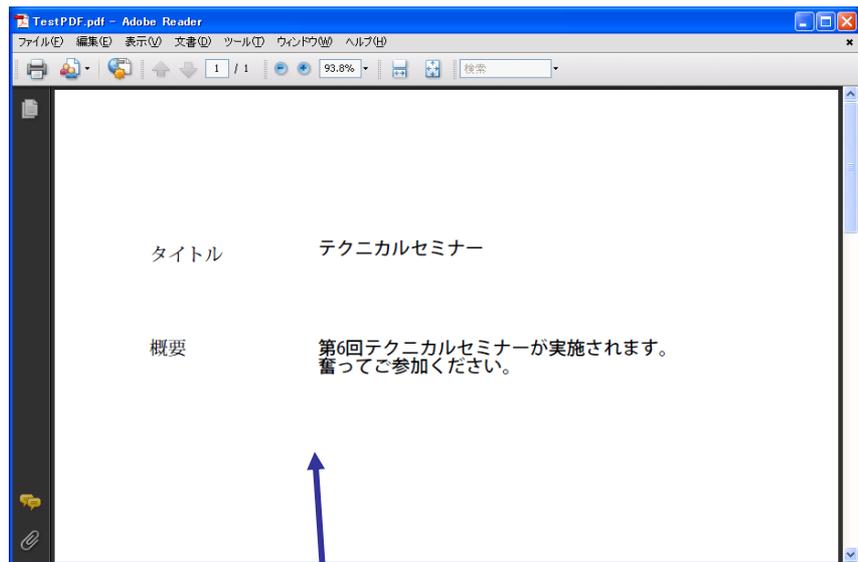
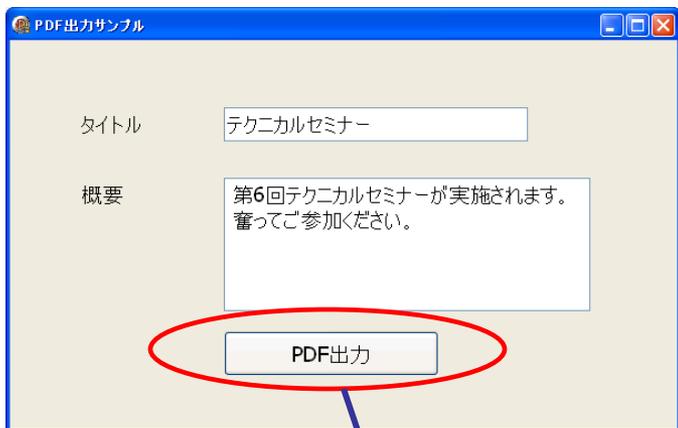
procedure TfrmSample.Button1Click(Sender: TObject);
begin
    // 保存ダイアログを表示
    if not SaveDialog1.Execute then Exit;

    frmPDFPrpt := TfrmPDFPrpt.Create(Self);
try
```

```
// レポート作成
with frmPDFPrpt do
begin
    // データをセット
    PRTText1.Text := Edit1.Text;
    PRTText2.Text := Memo1.Text;
    // PDF出力
    PReport1.FileName:=SaveDialog1.FileName;
    PReport1.BeginDoc;
    PReport1.Print(PRPage1);
    PReport1.EndDoc;
end;
finally
    frmPDFPrpt.Release;
end;
end;
```

## PowerPDF 開発手順

### ● 実行



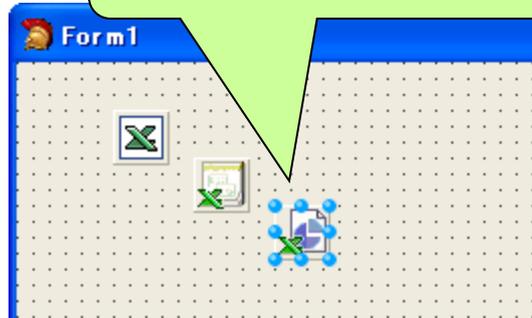
## ■Excel出力

### ● Excelファイルを生成する手法

- OLEによる出力
  - クライアント端末にExcelが必須
    - TExcelApplication
    - TExcelWorkBook
    - TExcelWorkSheet
- VB-Report (有償ツール)
  - エクセルをテンプレートとした帳票出力ツール
    - Delphiから操作するためのコンポーネント有 (V5~V2010対応)
    - クライアント端末は、Excel不要

※ VB-Report (ActiveX版) アドバンスソフトウェア  
[http://www.adv.co.jp/products/product\\_VB-Report3.htm](http://www.adv.co.jp/products/product_VB-Report3.htm)

プロパティ、メソッドはExcelのVBAを参照



[\*\*項目名]の形式でパラメータ指定

| No     | 商品コード  | 商品名    | 数量     | 単価     | 金額    |
|--------|--------|--------|--------|--------|-------|
| **SEQN | **PRCD | **PRNM | **PRSU | **PRIN | **PRI |
|        |        |        |        |        |       |
|        |        |        |        |        |       |
|        |        |        |        |        |       |
|        |        |        |        |        |       |
|        |        |        |        |        |       |

## ■ 将来を見越した帳票開発

### ● 帳票開発手法の変更

#### ● 現在QuickReportを使用

⇒ Delphiを最新版に変更する場合、QuickReport製品版にアップグレードするか、RaveReportsに**変更する必要あり**。

#### ● 現在の紙ベースの帳票を電子データに変更したい

⇒ RaveReoprts/QuickReportの帳票をPowerPDFやVB-Reportに**変更する必要あり**。



### ● 帳票出力手法の変更による改修範囲を小さくする工夫

⇒ **画面プログラムと帳票出力プログラムとを分離**するように留意する

## ■ 将来を見越した帳票開発

- 画面プログラムに直接帳票ロジックを埋め込むと...

RaveReportsの場合、(P.12参照)

```
procedure TfrmSample.Button1Click(Sender: TObject);
begin
  with RvProject1 do
  begin
    // Raveプロジェクトを開く
    Open;
    try
      // レポートの選択
      SelectReport('Report1', True);
      // 値のセット
      SetParam('Title', ...);
      SetParam('Summa...
```

PowerPDFの場合、(P.36参照)

```
procedure TfrmSample.Button1Click(Sender: TObject);
begin
  // 保存ダイアログを表示
  if not SaveDialog1.Execute then Exit;

  frmPDFPrpt := TfrmPDFPrpt.Create(Self);
  try
    // レポート作成
    with frmPDFPrpt do
    begin
      ...
```



開発手法を変更するたびに、画面プログラムのロジックも大きく変更しなければいけない

## ■ 将来を見越した帳票開発

### ● 考え方

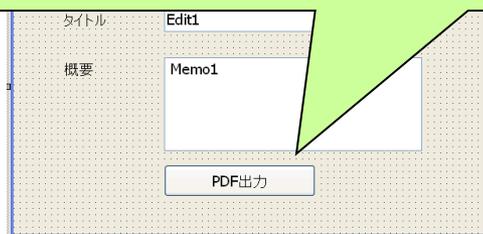
#### ● 帳票呼出ユニットを作成

- 帳票呼出に必要なパラメータを受け渡すプロパティ、帳票印刷するメソッドを作成
- 帳票プログラムと連動した出力ロジックを記述

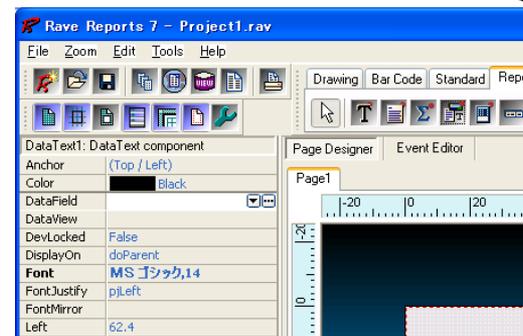
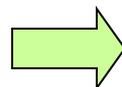
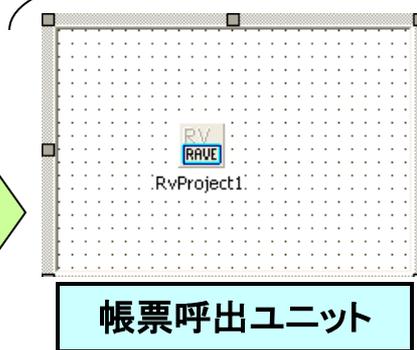
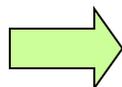
#### ● 画面プログラムは、帳票呼出ユニットの**プロパティ、メソッドにアクセス**

- 帳票呼出ユニットの仕様が変更されても**画面プログラムには影響しない**

帳票呼出ユニットのプロパティ、メソッドのみ使用。  
帳票コンポーネントは使用しない



画面プログラム



帳票プログラム

## ■ 帳票呼出ユニット実装例

### ● 帳票呼出ユニット

- プロパティ Title /Summary
  - 出力項目のセット用
- メソッド ReportPrint
  - 帳票出力処理

#### 宣言部

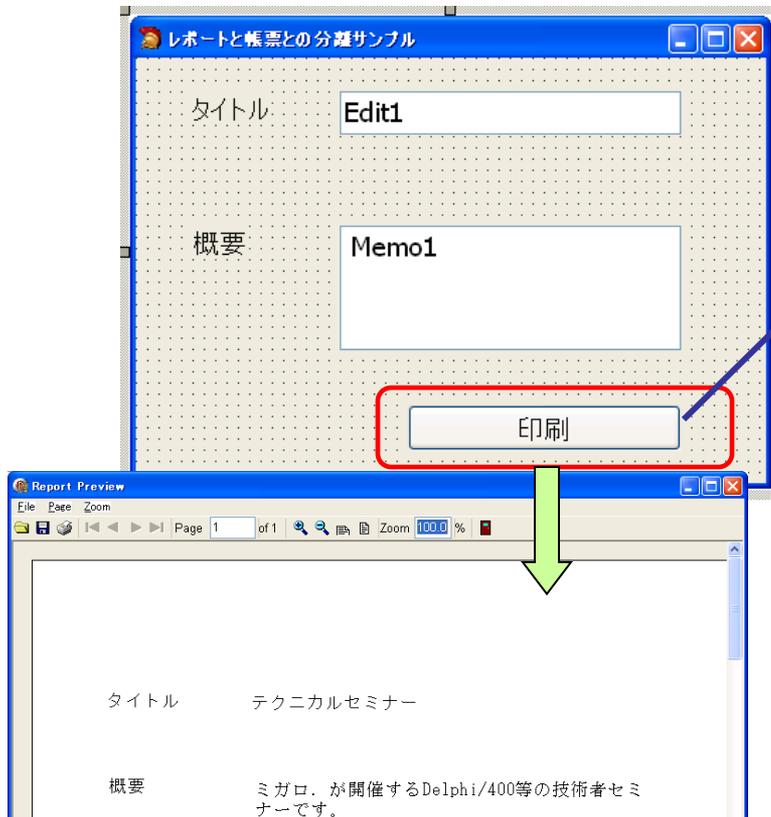
```
TdmReptModule = class(TDataModule)
  RvProject1: TRvProject;
private
  { Private 宣言 }
  FTitle: String; //---- タイトルをあらわす内部変数
  FSummary: String; //---- 概要をあらわす内部変数
public
  { Public 宣言 }
  property Title: String read FTitle write FTitle;
  property Summary: String read FSummary write FSummary;
  procedure ReportPrint;
end;
```

#### 実装部

```
procedure TdmReptModule.ReportPrint;
begin
  with RvProject1 do
  begin
    // Raveプロジェクトを開く
    Open;
  try
    // レポートの選択
    SelectReport('Report1', True);
    // 値のセット
    SetParam('Title', FTitle);
    SetParam('Summary', FSummary);
    // 実行
    Execute;
  finally
    // Raveプロジェクトを閉じる
    Close;
  end;
end;
end;
RaveReportsによる印刷ロジック
```

## ■画面プログラム帳票呼出実装例

### ●画面プログラム



```
uses ReportUnit;  
  
{$R *.dfm}  
  
procedure TfrmSample.Button1Click(Sender: TObject);  
begin  
    //-----帳票出力処理  
    //パラメータの指定  
    dmReptModule.Title := Edit1.Text;  
    dmReptModule.Summary := Memo1.Text;  
    //印刷実行  
    dmReptModule.ReportPrint;  
end;
```

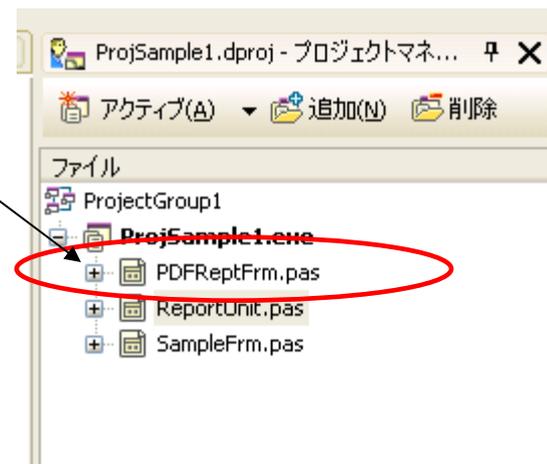
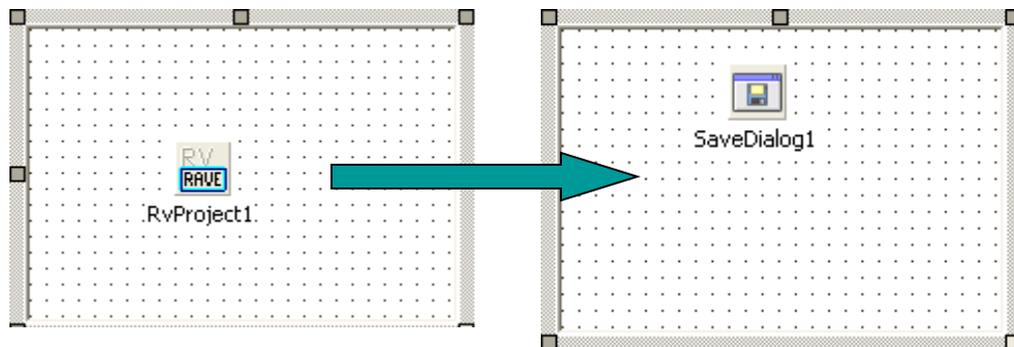
印刷指示画面上には、RaveReportsに関連するコンポーネントやメソッドは記述なし

実行時、RaveReportsにて出力

## ■ 帳票呼出ユニット仕様変更手順

### ● RaveReportsからPowerPDFに仕様変更

- RaveReportsで使用するコンポーネントを削除
  - RVProjectコンポーネント削除
- PowerPDFに必要なコンポーネントを追加
  - SaveDialogコンポーネント追加
  - PDF帳票プログラム(PDFReptFrm.pas)追加
- 印刷ロジック修正
  - 出力ロジック(ReportPrint)を改修



## ■ 帳票呼出ユニット仕様変更結果

### ● ソースコードの変更

- 帳票呼出ユニット変更
- ReportPrintメソッドのみ改修

#### PowerPDFによる出力ロジック

```
procedure TdmReptModule.ReportPrint;
begin
    // 保存ダイアログを表示
    if not SaveDialog1.Execute then Exit;

    frmPDFPrpt := TfrmPDFPrpt.Create(Self);
    try
        // レポート作成
        with frmPDFPrpt do
            begin
```

```
                // データをセット
                PRText1.Text := FTitle;
                PRText2.Text := FSummary;
                // PDF出力
                PReport1.FileName:=SaveDialog1.FileName;
                PReport1.BeginDoc;

                PReport1.Print(PRPage1);
                PReport1.EndDoc;
            end;
        finally
            frmPDFPrpt.Release;
        end;
end;
```

## ■仕様変更結果

### ● 画面プログラム



```
uses ReportUnit;  
  
{$R *.dfm}  
  
procedure TfrmSample.Button1Click(Sender: TObject);  
begin  
    //-----帳票出力処理  
    //パラメータの指定  
    dmReptModule.Title := Edit1.Text;  
    dmReptModule.Summary := Memo1.Text;  
    //印刷実行  
    dmReptModule.ReportPrint;  
end;
```

画面プログラム修正なし!



実行時、PDFファイルが出力

## ■ 帳票開発指針

### ● 帳票出力ツールの選択基準

- 標準ツールである“**RaveReports**”で対応可能か検討
  - リスト帳票であれば、データセットと連携可能な“RaveReports”が最適
  - 単票形式などであれば、“Excel出力”や“VB-Report”も有効な選択肢
- PDF出力を体系的に行う場合、“**PDFCreator**”の使用が有用
  - Delphiより直接PDF生成が可能
- “QuickReport”を既に利用の場合、製品版(Pro版)へのアップグレードか“RaveReports”への変更を検討
  - “RaveReports”は、“QuickReport”同様DataSetを使用可能
  - 帳票出力のパフォーマンス向上

### ● 帳票プログラム開発指針

- 画面プログラムと帳票プログラムとの分離を検討
  - 出力手法の変更が必要になった場合、画面プログラムに影響を及ぼさないことが可能(耐性の強いプログラムとなる)
  - 帳票プログラムの汎用化が可能(いろいろな画面から同じ帳票が汎用的に呼び出せる)

## システム開発事例 紹介

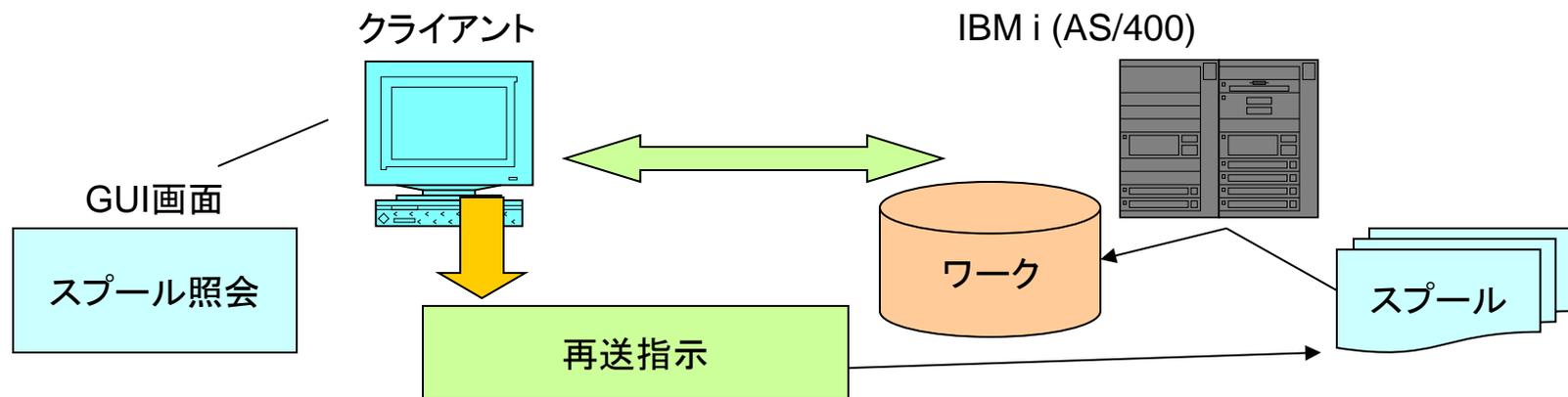
## ■AS/400スプールとの連携

### ● 案件概要 （自動FAX送信システム）

- ホスト(AS/400)を使用したFAX自動送信システムを使用しているが、送信エラーとなったものを簡単にチェックできないか？

⇒ AS/400のAPIと連携することにより簡単にスプールの情報を取り出せる照会画面をDelphiで構築

⇒ ステータスの変更を可能にすることで直接FAX再送が可能に。



## ■ AS/400スプールとの連携

### ● 開発事例紹介

| スプールファイル名 | ユーザー名      | ユーザー指定データ | 状況     | 合計ページ数 | 部数 | 用紙タイプ | スプールF番号      | ジョブ名 |
|-----------|------------|-----------|--------|--------|----|-------|--------------|------|
| QPQPRFIL  | D400       | 受注一覧      | *HELD  | 2      | 1  | *STD  | 1 QPADEV00H  |      |
| QPQPRFIL  | D400       | 売上月報      | *HELD  | 1      | 1  | *STD  | 2 QPADEV00H  |      |
| QPQPRFIL  | OKABAYASHI | 担当者別      | *READY | 1      | 1  | *STD  | 59 QPADEV00H |      |
| QPQPRFIL  | OKABAYASHI | 品目別       | *READY | 1      | 1  | *STD  | 60 QPADEV00H |      |
| QPQPRFIL  | OKABAYASHI | 得意先別      | *READY | 1      | 1  | *STD  | 61 QPADEV00H |      |

印刷装置: PRT052  
コピー枚数: 1  
印刷の再始動: 1  
保管ファイル: \*NO

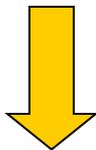
| ファイル    | QPQPRFIL | ジョブ名  | QPADEV00H | ユーザー    | D400    |           |       |              |       |    |    |
|---------|----------|-------|-----------|---------|---------|-----------|-------|--------------|-------|----|----|
| 10-0001 | 101      | 00001 | 山本        | 9/08/15 | 9/09/07 | 10000-000 | 東京工業機 | *** 受注一覧 *** | 受注行No | 1  | 1C |
| 10-0002 | 101      | 00002 | 青木        | 9/08/15 | 9/09/06 | 20000-000 | 浜松工業機 | 備考           | 2     | 1C |    |
| 10-0003 | 101      | 00001 | 山本        | 9/08/16 | 9/09/18 | 10000-000 | 東京工業機 | 納期前日にチェック    | 3     | 2C |    |
| 10-0004 | 101      | 00001 | 山本        | 9/08/17 | 9/09/01 | 10000-000 | 東京工業機 |              | 0     |    |    |
| 10-0005 | 101      | 00001 | 山本        | 9/08/18 | 9/09/08 | 10000-000 | 東京工業機 |              | 0     |    |    |
| 10-0006 | 102      | 00003 | 内野        | 9/08/20 | 9/09/10 | 30000-000 | 大阪工業機 |              | 0     |    |    |
| 10-0007 | 102      | 00003 | 内野        | 9/08/21 | 9/09/08 | 50000-000 |       | 納期前日にチェック    | 0     |    |    |
| 10-0008 | 101      | 00001 | 山本        | 9/08/21 | 9/09/08 | 10000-000 | 東京工業機 |              | 0     |    |    |
| 10-0009 | 101      | 00001 | 山本        | 9/08/21 | 9/09/08 | 10000-000 | 東京工業機 |              | 0     |    |    |

スプール状況の変更が可能

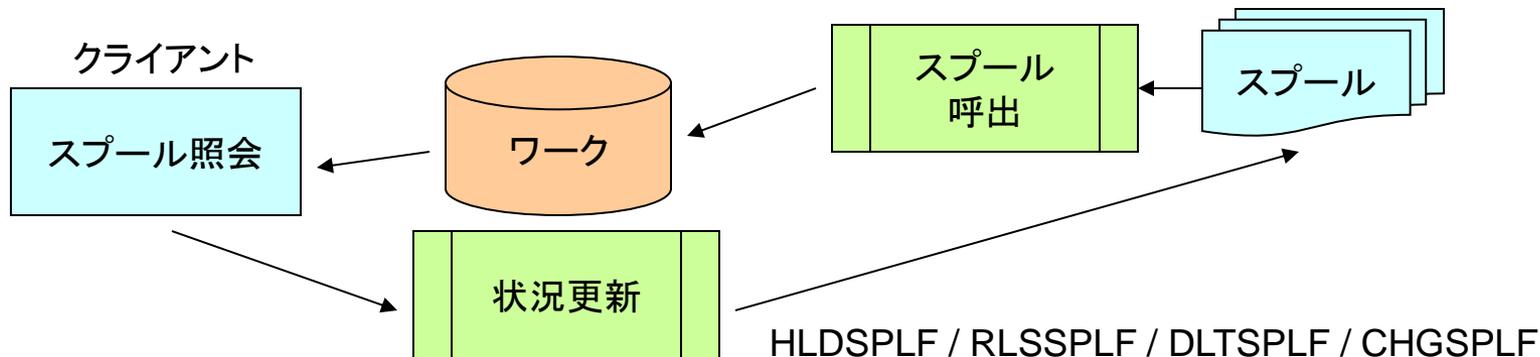
## ■AS/400スプールとの連携

### ● 開発手法

- ホスト(AS/400)を使用せずにスプールの再処理ができないか



- AS/400のAPIを使用することで、スプール内容をDB化
- CLを連動することにより、スプール状況の変更を可能に



※ APIを利用した出力待ち行列情報に関する詳細は、「ミガロ テクニカルレポート No1」参照

## ■ AS/400スプールとの連携

### ● Delphi使用のメリット

- ホストにログオンせずとも状況把握が可能
- 常駐プログラムとすることで、即時に状況確認が可能
- Delphi帳票プログラムとして、リスト出力が可能



### ● 活用事例

- スプールエラー時の自動メール配信
- スプールのファイル化することによる履歴管理

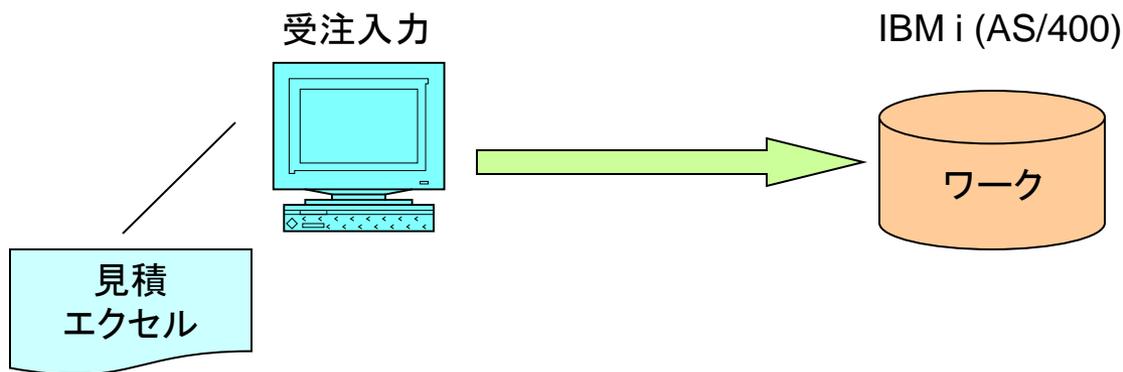
## ■ドラッグ & ドロップによる簡便なGUI操作

### ● 案件概要 (受発注システム)

- 別システムで作成した見積エクセルファイルを使用しているが、受発注システムに容易に連携できないか？

⇒ OLE処理によりエクセルから情報取得を可能に

⇒ ドラッグ & ドロップによる直感的操作にて簡便なシステムを構築



## ■ドラッグ & ドロップによる簡便なGUI操作

### ● 開発事例紹介

OLEを使用してエクセルを読み込み、画面項目にセット

Excelファイルをドラッグ

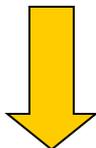
Excelファイルをドロップしてください。

| 分類 | 商品CD | 担当CD | 製品名                    | 数量 | 単価    | 金額     |
|----|------|------|------------------------|----|-------|--------|
| A  | A101 |      | 2 PlateSpin Forge      | 1  | 1,000 | 1,000  |
| B  | B102 |      | 29 Delphi/400 Ver.20   | 2  | 1,100 | 2,200  |
| A  | A103 |      | 7 JC/400 Ver.2010      | 3  | 1,200 | 3,600  |
| B  | B104 |      | 66 MKS Implementer     | 4  | 1,300 | 5,200  |
| A  | A105 |      | 71 *noMAX DEFENDER     | 5  | 1,400 | 7,000  |
| B  | B106 |      | 47 Delphi/400 Ver.2009 | 6  | 1,500 | 9,000  |
| A  | A107 |      | 41 UpdateObjects/400   | 7  | 1,600 | 11,200 |
| B  | B108 |      | 71 *noMAX GARRISON     | 8  | 1,700 | 13,600 |
| A  | A109 |      | 50 JC/400 Ver.4        | 9  | 1,800 | 16,200 |
| B  | B110 |      | 29 Simple Transfer/400 | 10 | 1,900 | 19,000 |

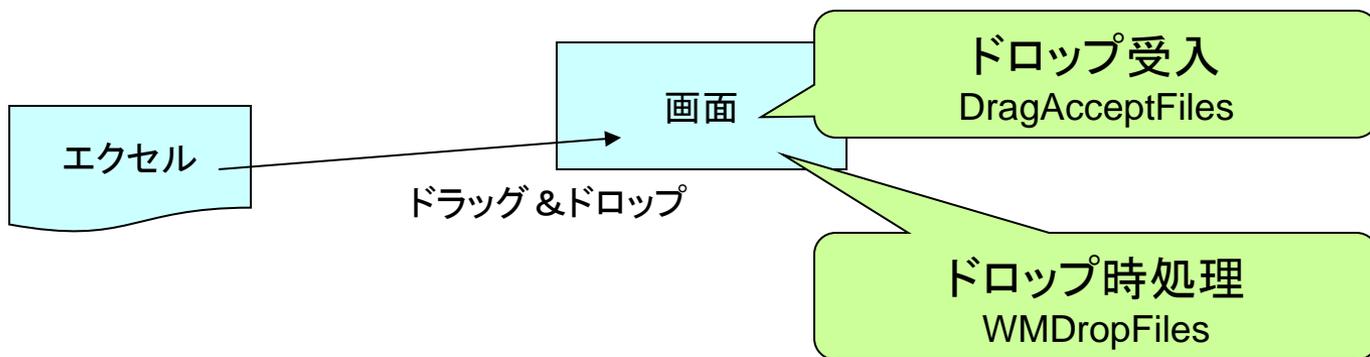
## ■ドラッグ & ドロップによる簡便なGUI操作

### ● 開発手法

- 見積書のエクセルを使って、受注入力の省力化はできないか？



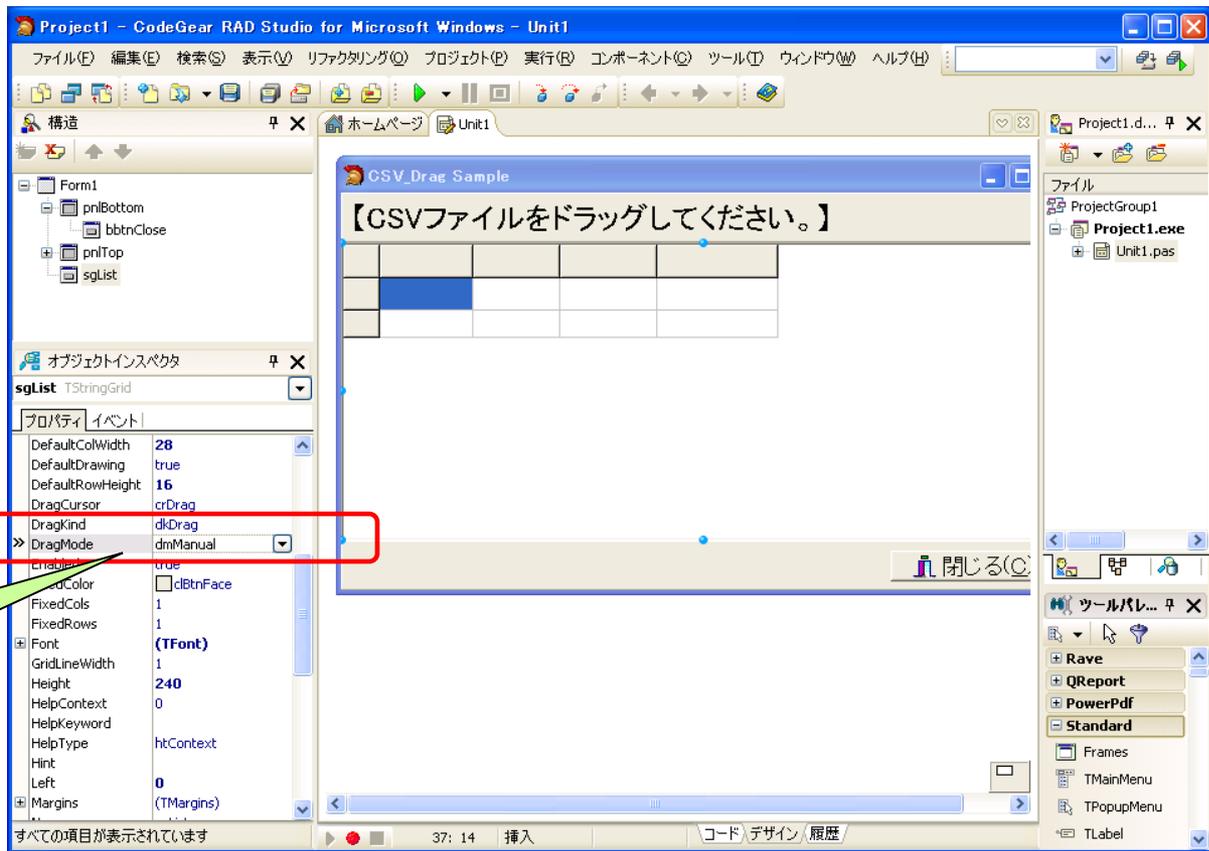
- OLEを使用することで、エクセルデータをプログラムで使用
- DragAcceptFiles手続きおよびWM\_DROPFILESメッセージによるドロップ処理の実現



## ■ドラッグ & ドロップによる簡便なGUI操作

### ● サンプルプログラムの作成

StringGridを配置



DragMode :=  
dmManual に設定

## ■ドラッグ & ドロップによる簡便なGUI操作

### ● サンプルプログラムの作成

```
type
  TForm1 = class(TForm)
    pnlTop: TPanel;
    lblTitle: TLabel;
    sgList: TStringGrid;
    pnlBottom: TPanel;
    bbtnClose: TBitBtn;
    procedure FormCreate(Sender: TObject);
  private
    { Private 宣言 }
    procedure WMDropFiles(var msg: TWMDropFiles);
    message WM_DROPFILES;
  public
    { Public 宣言 }
  end;
```

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  //ドロップをメッセージ可能にする
  DragAcceptFiles(Handle, True);
end;
```

ドロップ時発生するメッセージ

ドロップメッセージを使用可能にする

## ■ドラッグ & ドロップによる簡便なGUI操作

- サンプルプログラムの作成

```
procedure TForm1.WMDropFiles(var msg: TWMDropFiles);
```

```
var
```

```
  FileName : array[0..255] of Char;
```

```
  sFileName: String;
```

```
  slCSV1   : TStringList;
```

```
  slCSV2   : TStringList;
```

```
  ij : Integer;
```

```
begin
```

```
  try
```

```
    //ファイル名を取得
```

```
    DragQueryFile(msg.Drop, 0, FileName, SizeOf(FileName));
```

```
    sFileName := FileName;
```

```
  finally
```

```
    //ドラッグ完了時、ハンドルの解放
```

```
    DragFinish(Msg.Drop);
```

```
end;
```

ドロップされたファイルを  
取得

```
slCSV1 := TStringList.Create;
```

```
slCSV2 := TStringList.Create;
```

```
//ファイルのオープン
```

```
slCSV1.LoadFromFile(sFileName);
```

```
sgList.RowCount := slCSV1.Count; //行数取得
```

```
for i := 0 to sgList.RowCount - 1 do
```

```
begin
```

```
  slCSV2.CommaText := slCSV1[i]; //一行分情報
```

```
  sgList.ColCount := slCSV2.Count; //列数取得
```

```
  for j := 0 to slCSV2.Count - 1 do
```

```
    sgList.Cells[j,i] := slCSV2[j];
```

```
end;
```

```
slCSV2.Free;
```

```
slCSV1.Free;
```

```
sgList.Col := 1;
```

```
sgList.Row := 1;
```

```
end;
```

## ■ドラッグ & ドロップによる簡便なGUI操作

### ● サンプルプログラムの実行

CSVファイルをドラッグ

CSV内容が画面にセット

| 区分 | コード  | 品名          | 数量 | 単価   | 金額    |
|----|------|-------------|----|------|-------|
| B  | B104 | MKS Imp14   | 4  | 1300 | 5200  |
|    | A105 | *noMAX D5   | 5  | 1400 | 7000  |
| B  | B106 | Delphi/46   | 6  | 1500 | 9000  |
| A  | A107 | Update0b7   | 7  | 1600 | 11200 |
| B  | B108 | *noMAX G.8  | 8  | 1700 | 13600 |
| A  | A109 | JC/400 V.9  | 9  | 1800 | 16200 |
| B  | B110 | Simple T.10 | 10 | 1900 | 19000 |

## まとめ

## ■『帳票』開発ノウハウ公開

### 1)「RaveReports」による帳票開発手法

Delphi/400の標準レポートツールの使用手順

### 2)PDF出力技法

PowerPDFによるPDF作成手順

### 3)将来を見越した帳票開発

帳票出力仕様変更に強いシステム開発手順

## ■システム開発事例紹介

### 4)AS/400スプールとの連携

スプール取得APIの活用

### 5)ドラッグ&ドロップによる簡便なGUI操作

操作性を重視したアプリケーション構築