

直感的に理解できるシステムを目指して —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
製造総括部 部長代行



豊鋼材工業株式会社
<http://www.yutaka-steel.co.jp/>

「鋼材のことならあらゆるニーズに応える企業」をモットーに、広島から沖縄までをカバーし、鋼材およびその他金属の加工、販売を行っている。伊藤忠丸紅鉄鋼・新日本製鐵系の会社である。

ソリューション導入の経緯と概要

豊鋼材工業株式会社の主体である溶断事業では、従来、生産に関するさまざまな情報が、紙の帳票による事後の実績入力や、あるいは担当者の経験等に基づく繁閑の判断と工程管理に依存していた。したがって営業部門、経営者、さらに製造部門内からも、工場の操業状況や受注製品の進捗・計画状況はブラックボックス化して、お客様対応のレスポンスやさまざまな判断に支障をきたしていた。

このような状況を打開するため、平成17年より工場内の無線LAN強化をはじめ、各設備へのWindows CE 端末、無線ハンディターミナル、およびラベルプリンタの配備と帳票へのバーコード出力等の基盤整備を開始した。【図1-1】

Delphi/400は、この基盤整備でリアルタイムに収集される情報を定量的かつ直感的に利用する手段として導入した。本アプリケーションは、生産・出荷計画、進捗、履歴、山積み等の情報をさまざま

な角度から提供する参照系と、生産指示や計画作成等の更新系の処理を備えている。

開発の独創性・創意工夫

新しいシステム基盤Delphi/400等の導入により、従来は参照できなかった情報がリアルタイムに抽出可能となった。しかし、膨大な情報の中から、ユーザーが着目すべき部分をわかりやすく、かつ直感的に把握・操作できるようにしなければ、使い勝手のよいシステムとは言えない。

そのため、今回のアプリケーションでは次のような工夫している。

- ① DBGrid 表示データの中で着目すべき部分を着色表示、その閾値や色をユーザーによるカスタマイズ
- ② DBGrid の着色（色相）情報を VB-Report 色番号に変換し、Excel に出力（VB-Report）
- ③ DBGrid の選択行全体を強調表示

- ④ Chart を使用したトレンド等の可視化と、DBGrid 等との組み合わせによる詳細確認機能
- ⑤ DHTML 形式の操作マニュアルによる、シミュレーション体験型ヘルプ機能
- ⑥ Web ブラウザの活用
 - ・ GoogleMap によるお客様地図の表示（縮尺の異なる2段階表示）
 - ・ ネットワークカメラによる簡易 Web 会議
 - ・ ネットワークカメラによる工場の状況確認
 - ・ 無料の Web 型リモートデスクトップソフトによる画面共有
- ⑦ BitBtn、SpeedBtn 等への画像表示
- ⑧ DecisionGrid、DecisionGraph の活用

見える化への改善ポイント

アプリケーション開発の独創性・創意工夫について詳細を述べる。

図1-1 生産管理システム再構築イメージ

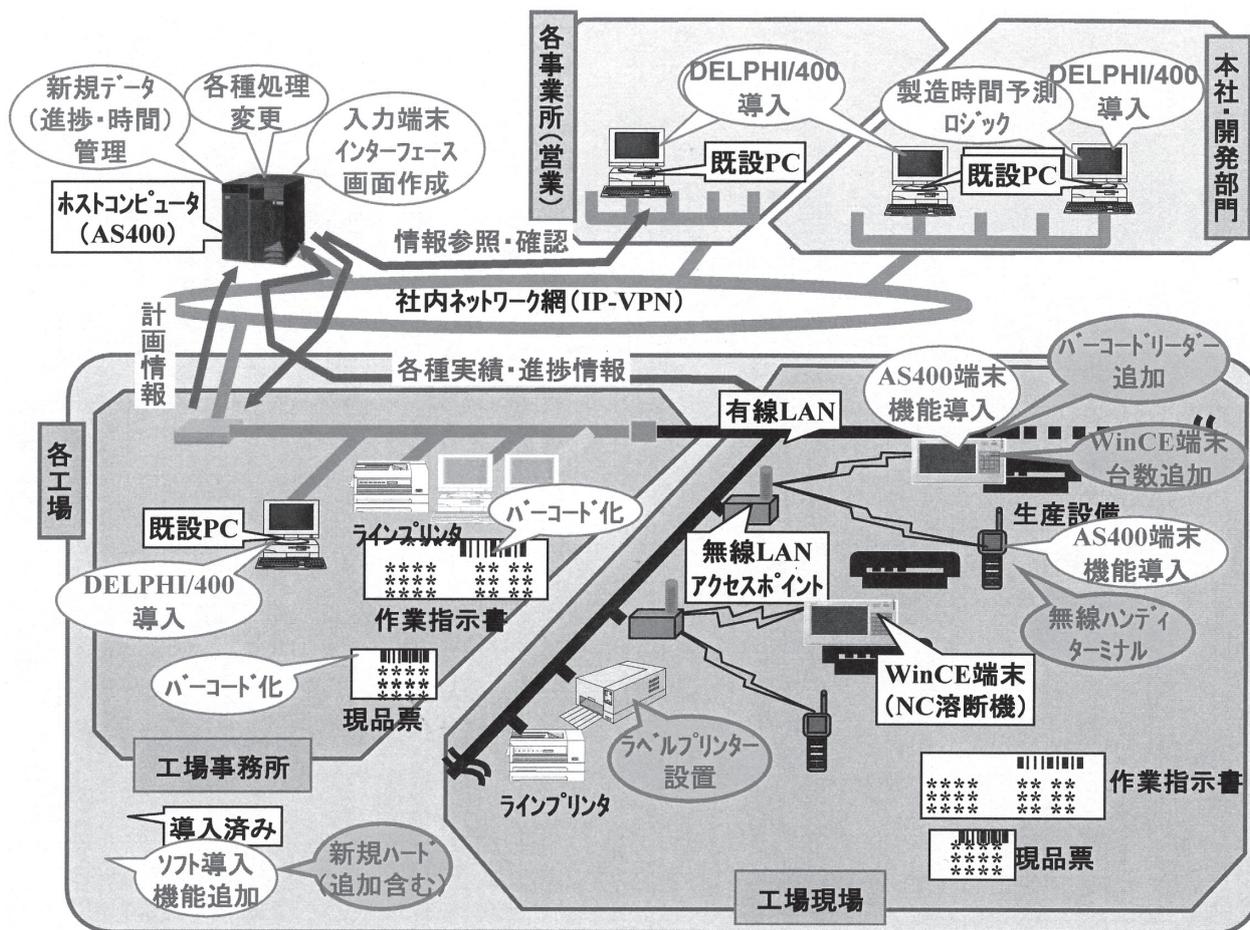


図2-1 汎用のGrid着色関数呼び出し例

```

procedure TForm1.PJJuSumGridDrawColumnCell(Sender: TObject;
  const Rect: TRect; DataCol: Integer; Column: TColumn;
  State: TGridDrawState);
var
  n: integer;
begin
  n := (Sender as TDbGrid).Tag;

  with (Sender as TDbGrid).Canvas do
  begin

    if GetGridColorForQP(n, DataCol) <> 0 then
      Brush.Color := GetGridColorForQP(n, DataCol);

    FillRect(Rect);
  end;

```

DBGridの配列番号=n、対象列番号=DataColで色を取得

① DBGrid の表示データ中で着目すべき行、セル、列あるいは特定のデータ内容などに応じて、傾向管理や識別容易化のために、OnDrawColumnCell イベントで着色判定用関数を呼び出している。

この関数は、呼び出し元の DBGrid の配列番号と項目を引数とし、DBGrid ごとの判定ロジックにより着色要否や色番号を返す。【図 2-1】【図 2-2】【図 2-3】

これによりユーザーは、着目すべき行、値の傾向等を直感的に把握可能になる。着色の判定の閾値と色には、ユーザーが入力した Edit の値や ColorBox で設定した色により変更可能なものもある。【図 2-4】

② VB-Report の機能で、Excel 出力する際に VB-Report3.0 の色定数を指定する必要がある。これは、Delphi 側の色を VB-Report3.0 の色定数に対応させる。【図 2-5】

DBGrid 各セルの着色用関数から返される色の色相範囲に対応して、VB-Report3.0 の色定数に変換する関数を作成した。【図 2-6】【図 2-7】

これにより、共通の着色ロジックでほぼ同等に着色された EXCEL の表を得ることが可能となった。

③ 通常 DBGrid では、選択行の左端の三角マークと選択セル色の反転で識別される。だが項目数が多い場合にはわかりにくい。

OnDrawColumnCell イベントで、TCustomDBGrid から派生させた TAccessDBGrid での選択行と現レコードのインデックスの一致判定により、着色を行うことが可能になった。【図 2-8】【図 2-9】

④ Query の動的 SQL で時系列に集約したデータを抽出し、ClientDataSet にキャッシュされた状態で、DBCrossTab Source 経由で DBChart 上の系列に接続している。

グループ化基準、集計項目等は、ユーザーの選択に応じて、DBCrossTab Source の GroupField、ValueField 等を動的に切り替える。ClientDataSet のキャッシュデータをもとにしているため、瞬時に角度を変えてのデータ分析が可能となった。

また、1つの Query の SQL で抽出されたデータに対して、DataSetProvider 経由で複数の ClientDataSet を接続し、各 ClientDataSet の Filter プロパティを個々に変える。1つの Query で、複数のグラフの一括表示を実現している。

【図 2-10】【図 2-11】【図 2-12】【図 2-13】【図 2-14】

さらに、グラフ上のダブルクリックで共通の TeeCommander を呼び出し、ユーザーが独自にグラフの書式編集、印刷、コピー等ができるようにした。その際、呼び出し元のグラフに応じて表示位置を変更し、共有化を実現している。【図 2-15】

グラフで全体感を確認した後に、系列のデータの内容をドリルダウンしたい場面も多い。グラフ上の該当部分をクリックすることで、その要素情報をパラメータとして詳細を表示する機能を多く実装した。【図 2-16】【図 2-17】【図 2-18】

⑤ TENDA 殿のマニュアル作成ツール (DOJO) の機能の 1つである、DHTML 形式の体験シミュレーション型マニュアルを各機能ページ別に作成し、社内の Web サーバーに保存した。

この該当 HTML を、各ページの Image のクリックイベントで、ShellExecute により呼び出せるようにした。【図 2-19】

実際に操作する感覚で機能・操作方法を習得できるため、学習効果が高く、マニュアル配布、教育等の負荷を軽減できる。音声解説も付加可能である。【図 2-20】【図 2-21】

なお、静的な HTML のマニュアル参照、PDF 形式のマニュアルのダウンロードも可能である。【図 2-22】【図 2-23】

⑥ Web ブラウザをコンポーネントとして組み込み、navigate メソッドで URL を指定するだけで容易に任意の Web ページを表示できる。この機能を応用して、お客様住所の GoogleMap 表示、ネットワークカメラ映像表示、リモートデスクトップ参照 (操作)【図 2-24】を可能とした。

・ GoogleMap

取得した Google Maps の APIkey を記述した雛形の HTML の住所とお客様

名の部分のみを、データベースとの連動により動的に置き換える。【図 2-25】【図 2-26】

GoogleMap の各種コントロールの組み込みは、Google Maps API の解説資料を参照した。

なお、地図ソフトでは目的地周辺の詳細と全体感を確認することが多いため、縮尺の異なる 2つの Map を 1つの URL で表示可能とし、印刷も可能とした。【図 2-27】【図 2-28】

住所情報が GoogleMap での検索に適合しない場合 (旧漢字等) は一部修正で、再試行可能とした。【図 2-29】

・ ネットワークカメラ

映像は、設置したカメラに付与した IP アドレスを URL で指定することで、見ることができる (Web サーバー機能内蔵による)。

今回、各カメラの名称、IP アドレス、アクセス権、接続可能時間等の基準をデータベース管理している。これにより、ユーザーに応じて使用条件をコントロール可能にし、プライバシー、ネットワーク負荷等へ配慮した。また、アクセス中の対象者がわかるように、アクセス状況もデータベース管理している。【図 2-30】

Web 会議と工場状況確認は、ComboBox でカメラを切り替えることで汎用的に使える。【図 2-31】

なお、Web 会議の際には、簡易な IP 電話会議機器を併用する。

⑧ユーザーのフォームからの項目選択により、任意の分析を可能とした。ただし、レコード件数が多い場合は、次元数によりメモリの使用量が大きくなり、実用レベルでは使えなくなる。項目選択数上限を設定している。【図 2-32】【図 2-33】

教訓・知見と今後の予定

Web ブラウザで Web 画面をコンポーネントとして組み込めるということ、Delphi を使い始めて 2 年近く気づかなかった。同様に、有効な機能を持つコンポーネントで知らないものが多くあると思われる。

また、標準では付属していないが、有効利用できる可能性があるものも確認できた。

図2-2 DBGridの着色関数の例

```
//Gridの着色を設定する汎用関数
function TForm1.GetGridColorForDP(n:integer;j:integer): integer;
var
  todayNum,todayNum2:integer;
  selcolor:Tcolor;
  colornum:integer;
begin
  colornum:=0; //初期値
  selcolor:=ClWhite; //初期値

  case n of
    36:
      begin
        if (Form1.mainCDS[n].FieldByName('SYU033T').Value=Form1.mainCDS[n].FieldByName('JU2013').Value) Or ((Form1.mainCDS[n].FieldByName('HAX008T').Value=Form1.mainCDS[n].FieldByName('JU2013').Value) then //全部配車
          begin
            selcolor := Form1.StateColorSelFin.Color;
          end
        else if Form1.mainCDS[n].FieldByName('HAX008T').Value=Form1.mainCDS[n].FieldByName('JU2013').Value then //全部配車
          begin
            selcolor := Form1.StateColorSelD.Color;
          end
        else if Form1.mainCDS[n].FieldByName('L02006F').Value=Form1.mainCDS[n].FieldByName('JU2013').Value then //全部生産
          begin
            selcolor := Form1.StateColorSelP.Color;
          end
        else if (Form1.mainCDS[n].FieldByName('L02006F').Value>0) And (Form1.mainCDS[n].FieldByName('L02006F').Value<Form1.mainCDS[n].FieldByName('L02006F').Value then //全部指示
          begin
            selcolor := Form1.StateColorSelSP.Color;
          end
        else if Form1.mainCDS[n].FieldByName('L02006T').Value=Form1.mainCDS[n].FieldByName('JU2013').Value then //全部指示
          begin
            selcolor := Form1.StateColorSelL.Color;
          end
        else //指示未
          begin
            selcolor := Form1.StateColorSelN.Color;
          end;
      end;
    end; //n=36の場合ここまで

    if selcolor<>ClWhite then
      colornum:=selcolor
    else
      colornum:=0;

    result:=colornum ;
  end;
```

呼び出したDBGrid毎に着色ロジックを記述

36:

図2-3 DBGridの着色の例

Y-VIS(MP) 一新生産・在庫管理メニュー

素材在庫関係(M) 生産・出荷状況(P)

担当: 物件等より抽出 | 受発注より抽出 | 工場単位で抽出

受注毎の製造・出荷状況

納期範囲: 2008/07/22 ~ 2008/08/05

15:08 DATE

自動更新: 更新時間: 15:04

一覧表示 | 計画情報 | 地図表示

生産・出荷明細不要 | 優先 | 待込先

生産指示済 | 生産指示済

生産一部済 | 生産済

配車済 | 出荷済

受発注別生産・出荷実績 (46 / 506)

地区GR	受発注行	部門	担当	客先	持込先	工場	売区	受区	納期	入力日	日外日	生産予定	完成時間	最終完成日	積込予定	配車最終日	出荷
福岡近郊	E 23575	13	鋼材課	相川	旭業	博多区諸岡5-16	330	11	1	80715	80708	80709	0	0	710	0	80723
福岡近郊	E 23577	1	鋼材課	相川	旭業	博多区諸岡5-16	430	11	1	80715	80708						80723
福岡近郊	E 23577	2	鋼材課	相川	旭業	博多区諸岡5-16	430	11	1	80715	80708						80723
筑後・鳥栖	E 23576	1	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80804	805	1508	0	0	
筑後・鳥栖	E 23576	2	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80804	805	1508	0	0	
筑後・鳥栖	E 23576	3	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80804	805	1426	0	0	
筑後・鳥栖	E 23576	4	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80804	805	1426	0	0	
筑後・鳥栖	E 23576	5	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80804	805	1426	0	0	
筑後・鳥栖	E 23576	6	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80804	805	1426	0	0	
筑後・鳥栖	E 23576	7	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80804	805	1426	0	0	
筑後・鳥栖	E 23576	8	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80804	805	1426	0	0	
筑後・鳥栖	E 23576	9	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80804	805	1426	0	0	
筑後・鳥栖	E 23576	10	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80805	805	1508	0	0	
筑後・鳥栖	E 23576	11	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80709	0	0	710	0	
筑後・鳥栖	E 23576	12	鋼材課	相川	旭業	市現場	330	11	1	80725	80708	80709	0	0	710	0	
筑後・鳥栖	E 23578	1	鋼材課	相川	旭業	市現場	430	11	1	80725	80708						
筑後・鳥栖	E 23578	2	鋼材課	相川	旭業	市現場	430	11	1	80725	80708						
熊本市近郊	E 24568	1	鋼材課	相川	旭		389	10	4	80722	80714						80722
熊本市近郊	E 23591	2	鋼材課	相川	旭		430	11	1	80724	80715						
熊本市近郊	E 25801	1	鋼材課	相川	旭		430	11	1	80725	80717	80722		722			80724
北九州近郊	F 43444	1	鋼材課	相川	ルワ	ニコム若松工場	330	13	1	80724	80718	80722	723	1006	723	0	80724
北九州近郊	F 43444	2	鋼材課	相川	ルワ	ニコム若松工場	330	13	1	80724	80718	80722	723	903	723	0	80724

受発注の明細単位での工程進捗状況を色分け

生産状況・原価明細 | 配車・出荷履歴明細

生産状況明細

選択のみ | 全明細

明細データ取込み設定されていません

これらについて Study し、レベルアップを図りたい。

エンドユーザー評価

●印刷帳票から Excel に手入力したデータをもとに定期的に作成していた（または、できなかった）指標の長期・短期の最新トレンドグラフが、人の手を煩わすことなく確認できる。

資料作成、データ取り込み等の手間も大幅に軽減できた。

●工程進捗状況が、DBGrid の色分けにより、データの値を見なくても確認できる。対応を迅速に行えるため非常に有効である。

M

図2-4 Grid着色の閾値、色を設定可能とした例

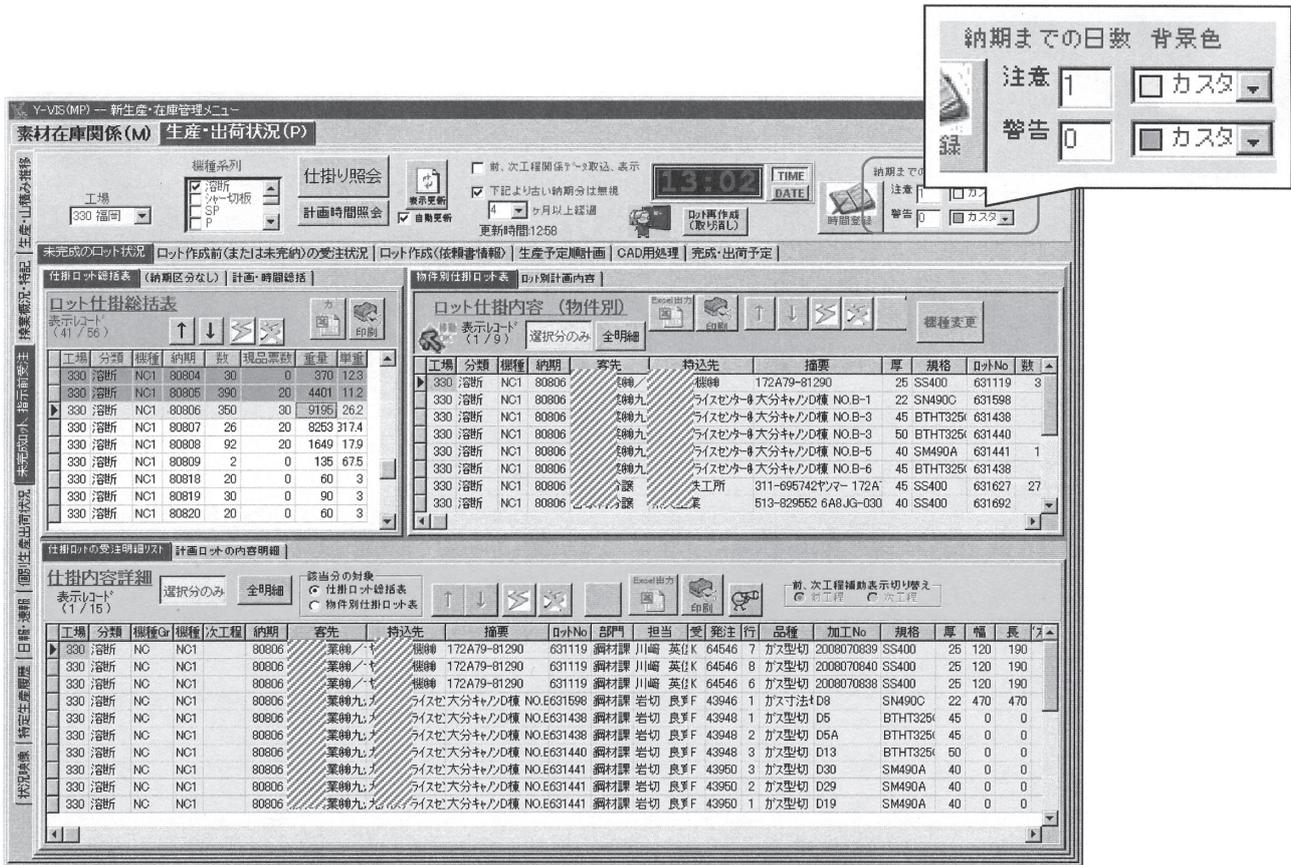


図2-5 VB-Report3.0の色番号例

設定値	定数	値	内容
	xcDefault	0	自動(デフォルト値)
	xcBlack	8	黒
	xcWhite	9	白
	xcRed	10	赤
	xcGreen	11	緑
	xcBlue	12	青
	xcYellow	13	黄色
	xcPink	14	ピンク
	xcCyan	15	水色

上記以外の色の設定する場合は、下記の値を参照してください。

16	濃い赤	17	暗い緑	18	濃い青
19	濃い黄	20	紫	21	青緑
22	25%灰色	23	50%灰色	24	グレー
25	ナラム	26	アイボリー	27	薄い水色
28	濃い紫	29	コーラル	30	オーシャンブルー
31	アイスブルー	32	濃い青	33	ピンク
34	黄色	35	水色	36	紫

図2-6 EXCEL出力用の色番号取得用関数呼び出し(抜粋)

```

while not mainCDS[j].Eof do //明細データ
begin
ii:=0;
for i:=0 to mainCDS[j].FieldCount-1 do
begin
if mainCDS[j].Fields[i].Visible=true then
begin
XlsReport1.Pos[ii,n,ii,n].Value:=mainCDS[j].Fields[i].Value;
XlsReport1.Pos[ii,n,ii,n].Attr.Box[0]:=lsNormal;

if GetGridColorForOP(j,ii)<>0 then
begin
if EXCELcolorConv(GetGridColorForOP(j,ii))<>999 then
XlsReport1.Pos[ii,n,ii,n].Attr.BackColor:=EXCELcolorConv(GetGridColorForOP(j,ii));
end;
ii:=ii+1;
end;
end;
n:=n+1;
mainCDS[j].next;
end;

```

対象DBGridの各セルの色を設定

対象DBGridの列毎の色をVB-Reportの色番号に変換

図2-7 同等の色相のVB-Report色番号に変換する関数

```

function TForm1.EXCELcolorConv(originColor: integer): integer;
var
h: Real; // 色相 hue (0~360) → この値で着色内容を設定
s: Real; // 彩度 saturation
v: Real; // 明度 value
min: Integer;
max: Integer;
delta: Integer;
IntColor: Integer;
r,g,b:Byte;
begin
IntColor:= ColorToRGB(originColor);
r:= GetRValue(IntColor);
g:= GetGValue(IntColor);
b:= GetBValue(IntColor);

ここにr,g,bからh,s,b(各色相、彩度、明度)
に変換する処理を記述

case round(h) of
0..30: result:=29;
31..60: result:=47;
61..90: result:=43;
91..120: result:=42;
121..150: result:=57;
151..180: result:=41;
181..210: result:=44;
211..240: result:=31;
241..270: result:=24;
271..300: result:=46;
301..330: result:=29;
331..360: result:=45;
else
result:=999;
end;
end;

```

色相(h)の範囲に応じてVB-Reportの色番号を割り当て

図2-8 DBGridの選択行着色例

SOZ092	SOZ003	SOZ105	SOZ014	SOZ015	SOZ004	HNA04K	SOZ005	SOZ006	SOZ007	SOZ008	SOZ102	SOZ101	
0000003	330	430	1	1	2100	生産定尺			2.3	914	1829	0	0
0000004	330	430	1	1	2100	生産定尺			2.3	1219	2438	0	0
0000005	330	430	1	1	2100	生産定尺			2.3	1524	3048	0	0
▶1752379	330	430	1	3	2100	生産定尺			2.3	1524	3048	0	0
0000006	330	430	1	1	2100	生産定尺			3.2	914	1829	0	0
0000007	330	430	1	1	2100	生産定尺			3.2			0	0
0000008	330	430	1	1	2100	生産定尺			3.2			0	0
0000010	330	430	1	1	2100	生産定尺			4.5	914	1829	0	0
0000011	330	430	1	1	2100	生産定尺			4.5	1219	2438	0	0



素材No	元コードNo	工場	分譲元	自支	規格	メーカー	品種	品名	板厚	幅	長さ	尺寸	W1	L1	数量	重量	橋梁表面	引当数	引当重	紐付部門	紐付客先
0000016	330	430	1			新日鐵	2100	生産定尺	6	1524	3048	5 x 10	0	0	80	17520		81	17739		0
1401115	330	430	1			新日鐵	2100	生産定尺	9	914	1829	3 x 6	0	0	99	11682		15	1770		0
▶1771622	330	0	1			新日鐵	2100	生産定尺	9	914	1829	3 x 6	0	0	1	118		0	0		0
0000021	330	430	1			新日鐵	2100	生産定尺	9	1219	2438	4 x 8	0	0	194	40740		5	1050		0
0483655	330	430	1			新日鐵	2100	生産定尺	9	1524	3048	5 x 10	0	0	1						0
1770845	330	0	1			新日鐵	2100	生産定尺	9	1524	3048	5 x 10	0	0							0
0000025	330	430	1			新日鐵	2100	生産定尺	12	914	1829	3 x 6	0	0	2						0
0000027	330	430	1			新日鐵	2100	生産定尺	12	1524	3048	5 x 10	0	0	3	1314		0	0		0
0000030	330	430	1			新日鐵	2100	生産定尺	16	914	1829	3 x 6	0	0	24	5040		24	5040		0
0000032	330	430	1			新日鐵	2100	生産定尺	16	1524	3048	5 x 10	0	0	10	5830		0	0		0

図2-9 選択行着色の処理コード部分抜粋

```

type
  TAccessDBGrid = class(TCustomDBGrid);
  TAccessDBGridを宣言

with TAccessDBGrid(Sender) do
  begin
    //選択行を着色したい
    //描画行と現在行が一致するとき色を設定
    if (DataLink.ActiveRecord = (Row-1)) then //ActiveRecordは描画している行(0から). Rowは選択行(1から)
      //対象列を指定
      begin
        if column.Field=mainDBGrd[n].SelectedField then
          mainDBGrd[n].Canvas.Brush.Color := clLtGray ;
          mainDBGrd[n].Canvas.Font.Color := clBlue;
          mainDBGrd[n].Canvas.FillRect(Rect);
        end;
      end;
    end;
    mainDBGrd[n].DefaultDrawColumnCell(Rect,DataCol,Column,State);
  } (OnDrawColumnCell内に記述)
  
```

図2-10 複数種グラフ連続作成のためのコンポーネント接続イメージ

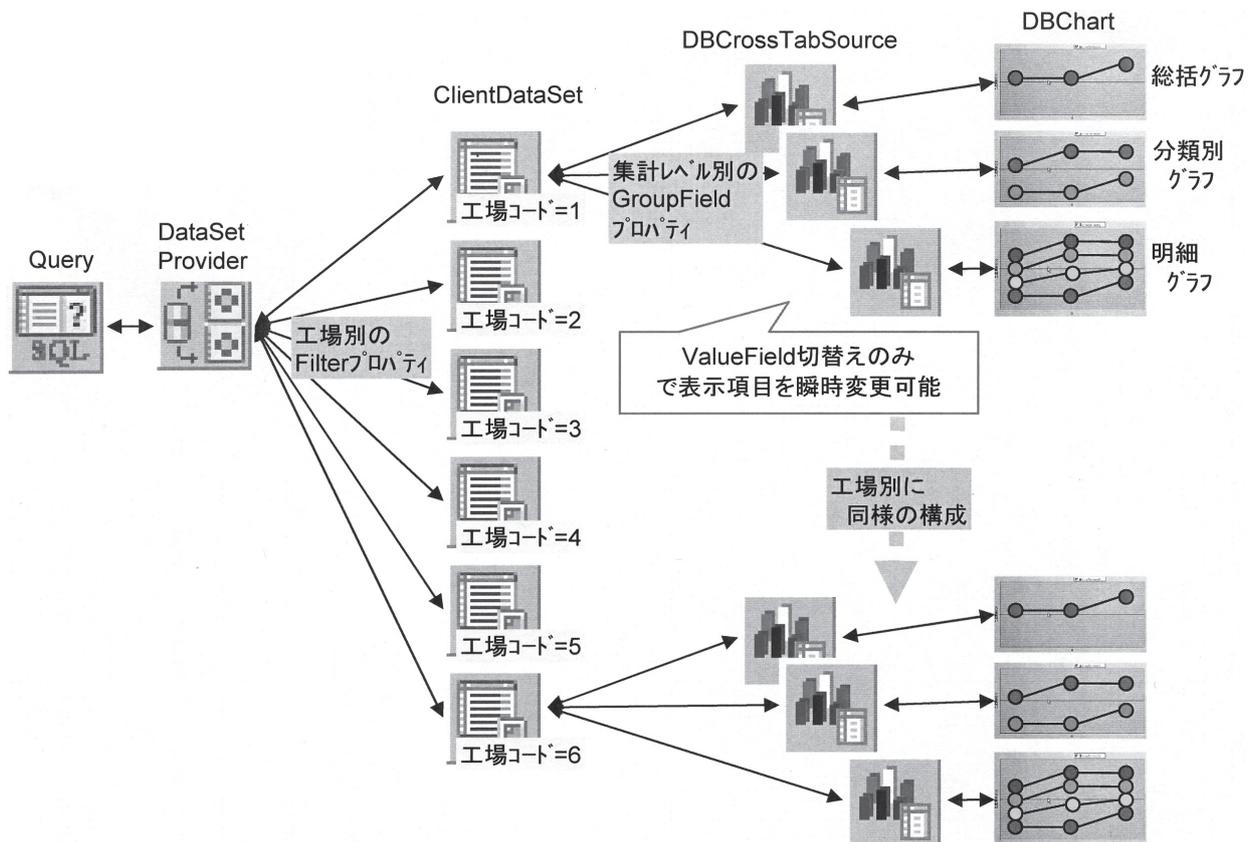


図2-11 複数種グラフ連続作成の基本処理部分

```

for i:=1 to High(DailyPTrendCDS) do //工場別のClientDataSet毎に処理
begin
DailyPTrendCDS[i].Filtered:=False;
DailyPTrendCDS[i].Active:=False;

DailyPTrendCDS[i].Active:=True; //各ClientDataSetをOPENしなおす、フィルターも再度有効化
DailyPTrendCDS[i].Filtered:=True;

for j:=1 to 3 do //総括、グループ別、明細の3レベルで作成
begin
DailyPTrendTabSrc[i,j].Active:=false; //各DBCrossTabSourceは工場、集計レベル毎の2次元配列
DailyPTrendTabSrc[i,j].Active:=true; //各DBCrossTabSourceをActiveにしなおす
//ここにグラフ描画等に関する各種処理を記述
end;
end;
    
```

図2-12 複数種グラフの表示項目連続切替えの処理例

```

for i:=1 to High(DailyPTrendCDS) do //工場別のClientDataSet毎に処理
begin
DailyPTrendCDS[i].Filtered:=False;

for j:=1 to 3 do //総括、グループ別、明細の3レベルで作成
begin
DailyPTrendTabSrc[i,j].Active:=false;
DailyPTrendTabSrc[i,j].ValueField:=hyoujitem; //予め設定した項目(ここではhyoujitem)に変更
DailyPTrendTabSrc[i,j].Active:=true; //各DBCrossTabSourceをActiveにしなおす
//ここにグラフ描画等に関する各種処理を記述
end;
end;
    
```

図2-13 グラフ一括表示の例

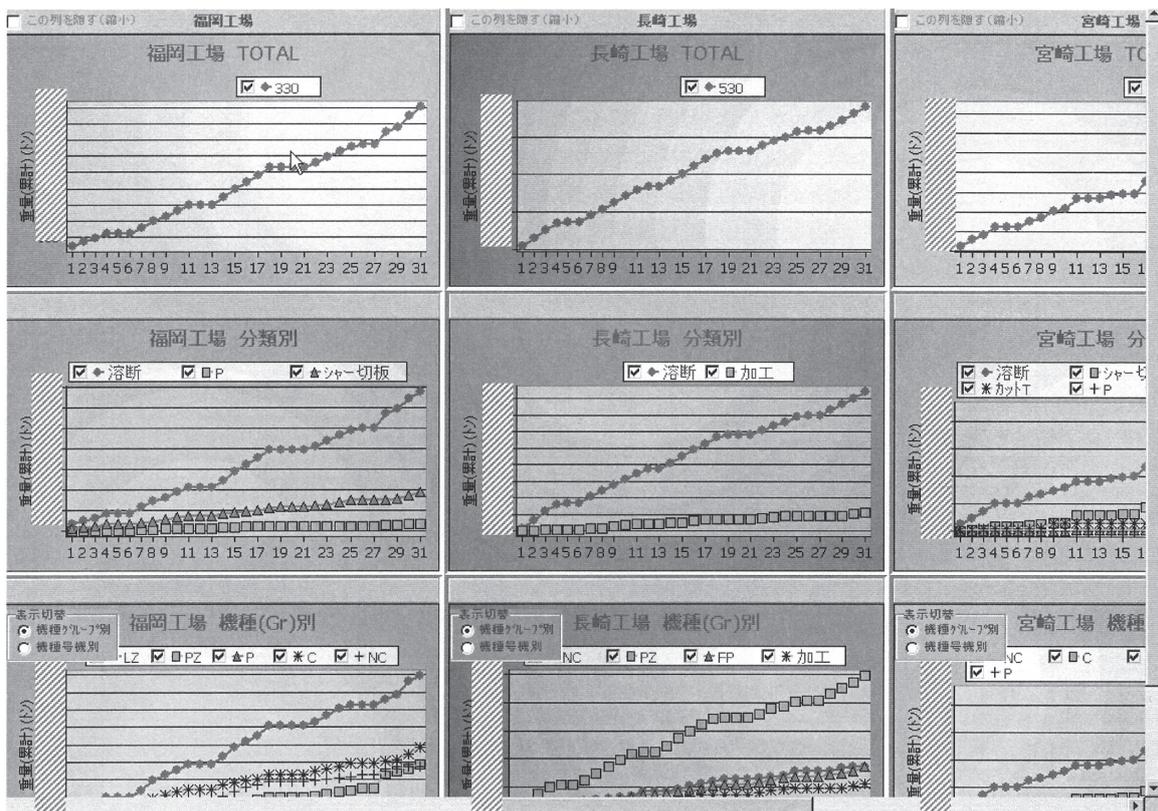


図2-14 グラフの表示項目一括変更の例

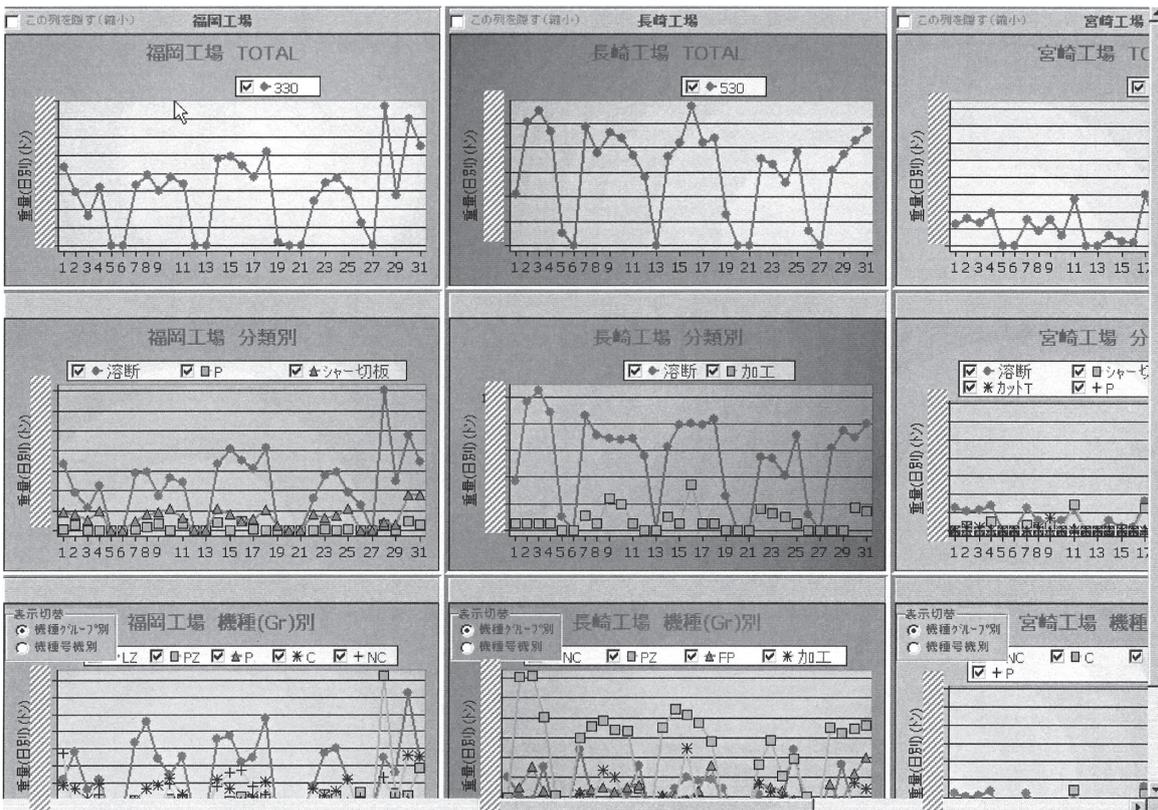


図2-15 TeeCommander呼び出しの例

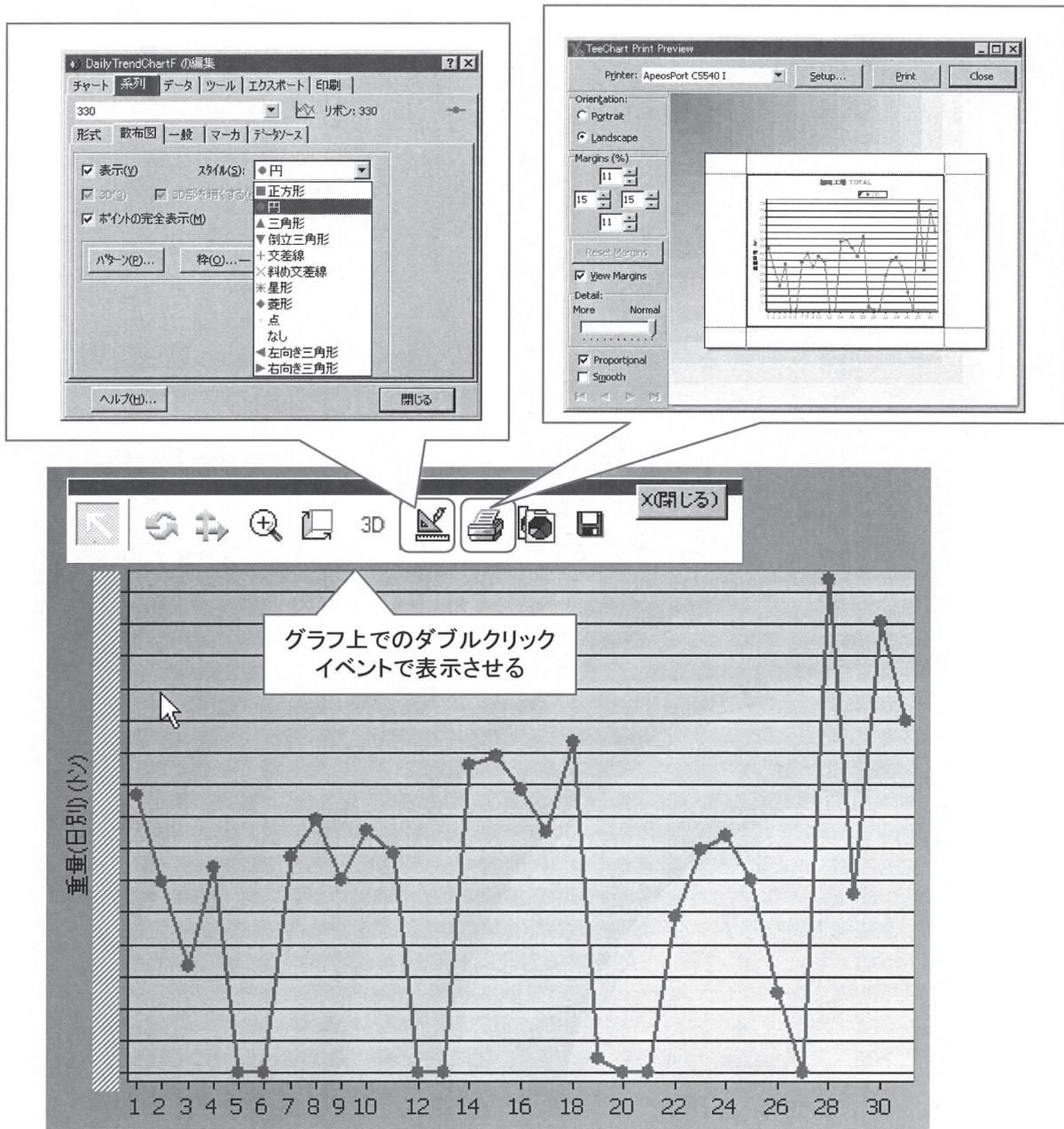


図2-16 系列要素クリックでのデータ表示例-1

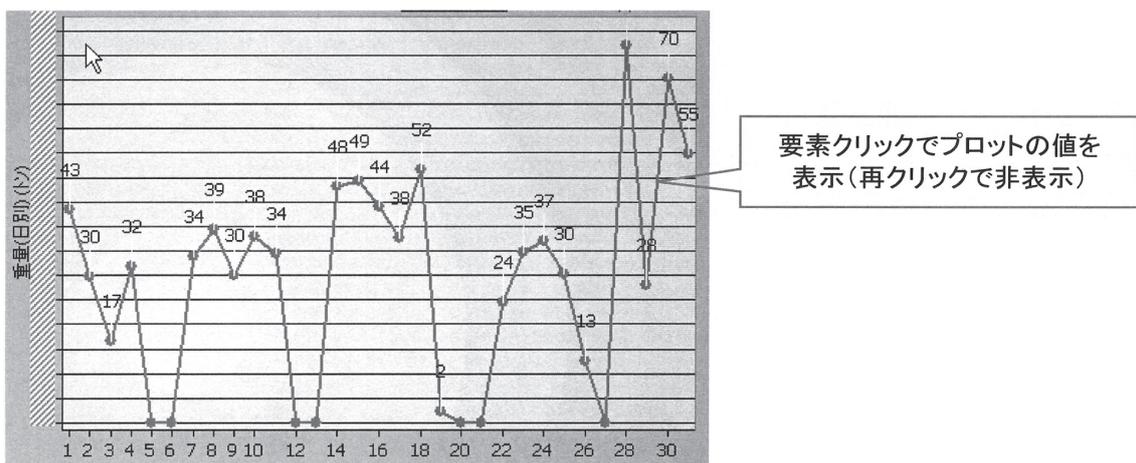


図2-17 系列要素クリックでのデータ表示例-2

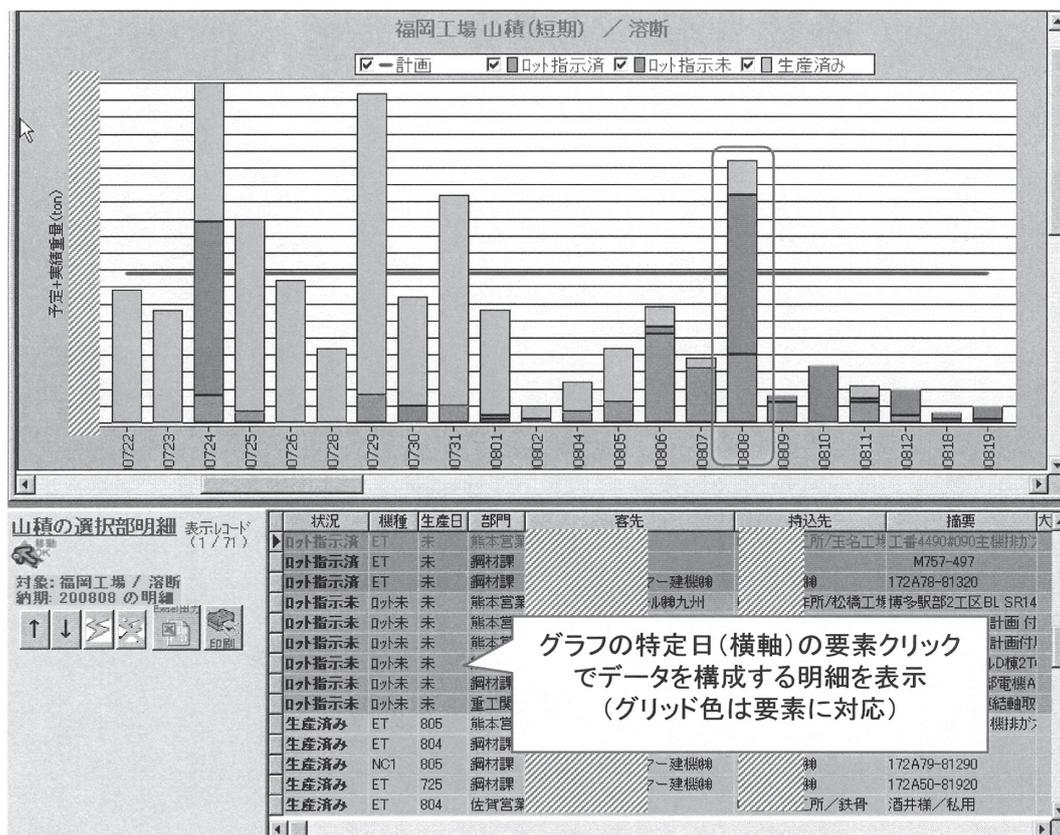


図2-18 系列要素クリックでのデータ表示例-3

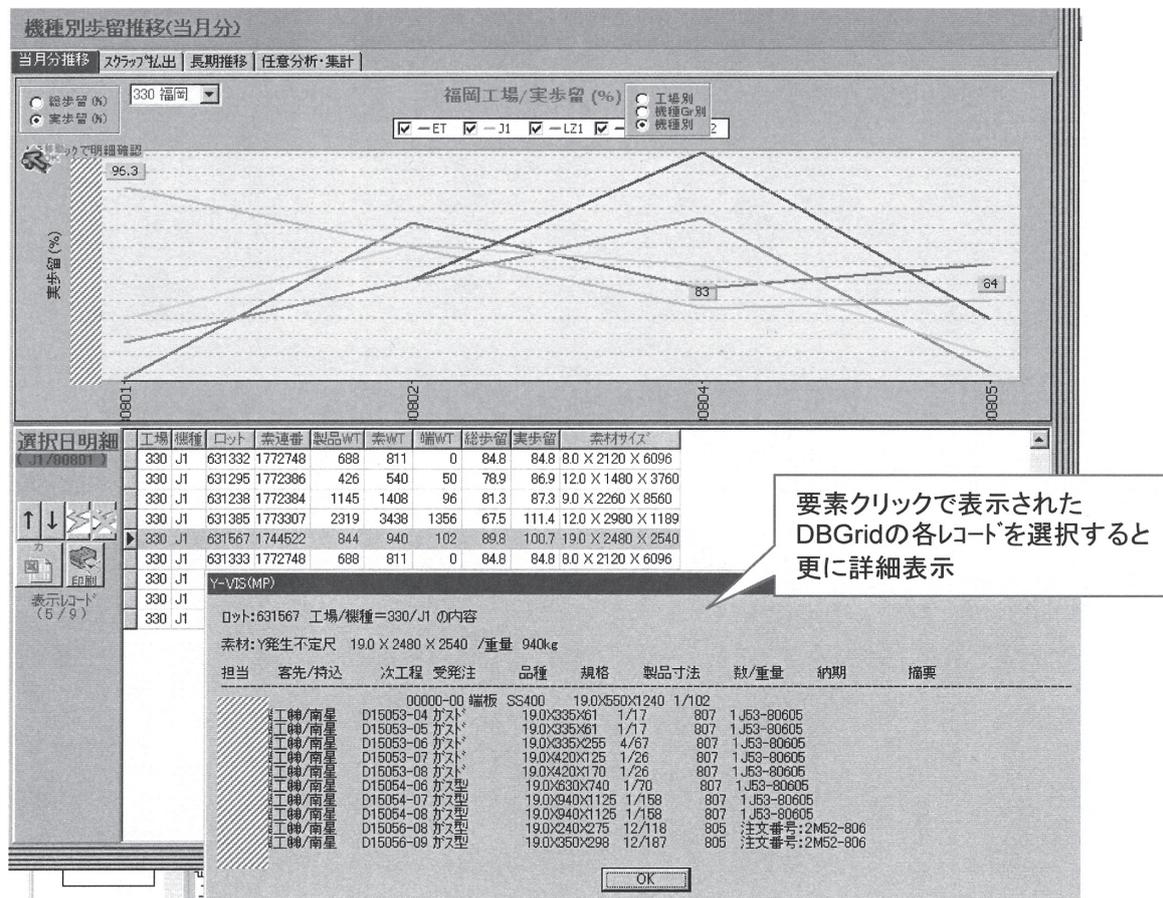


図2-19 マニュアル呼び出し方法例

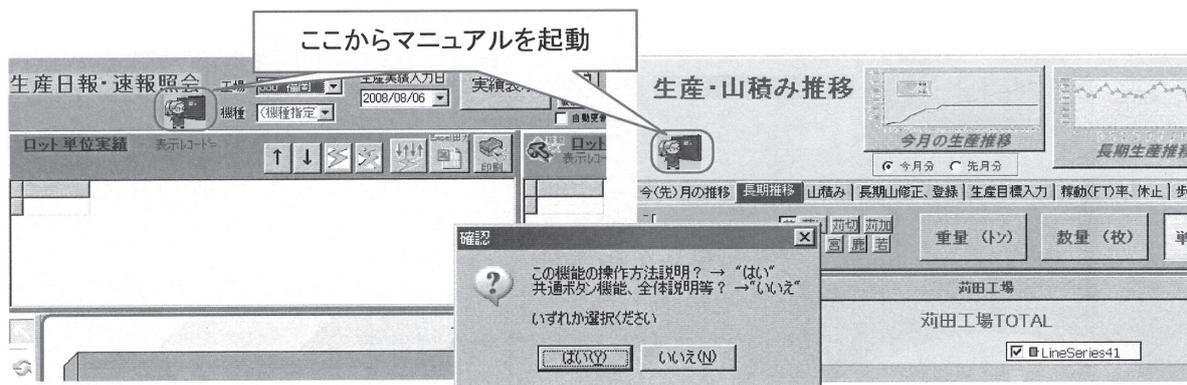


図2-20 DHTML形式のシミュレーション型マニュアル初期画面例

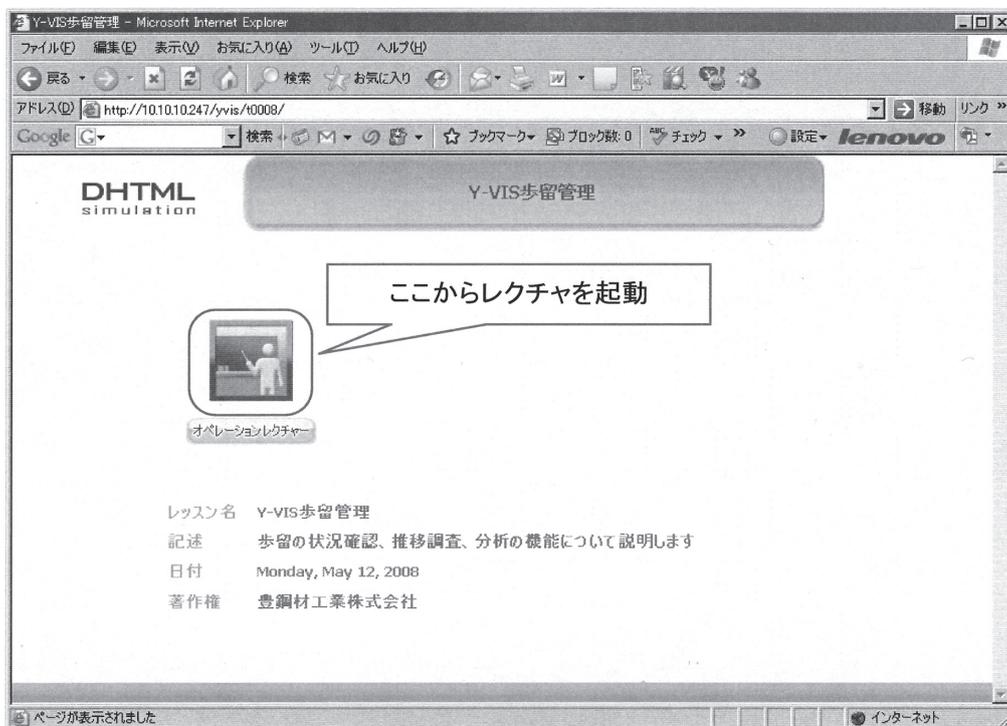


図2-21 DHTML形式のシミュレーション型マニュアル操作例

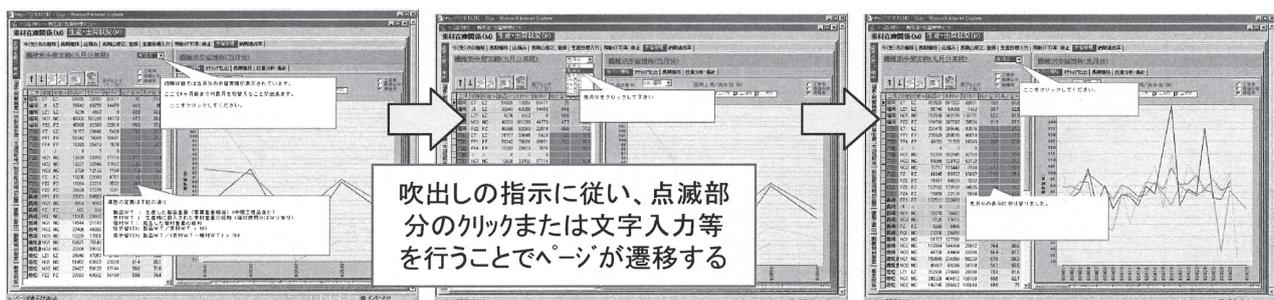


図2-22 静的HTML形式のマニュアル初期画面

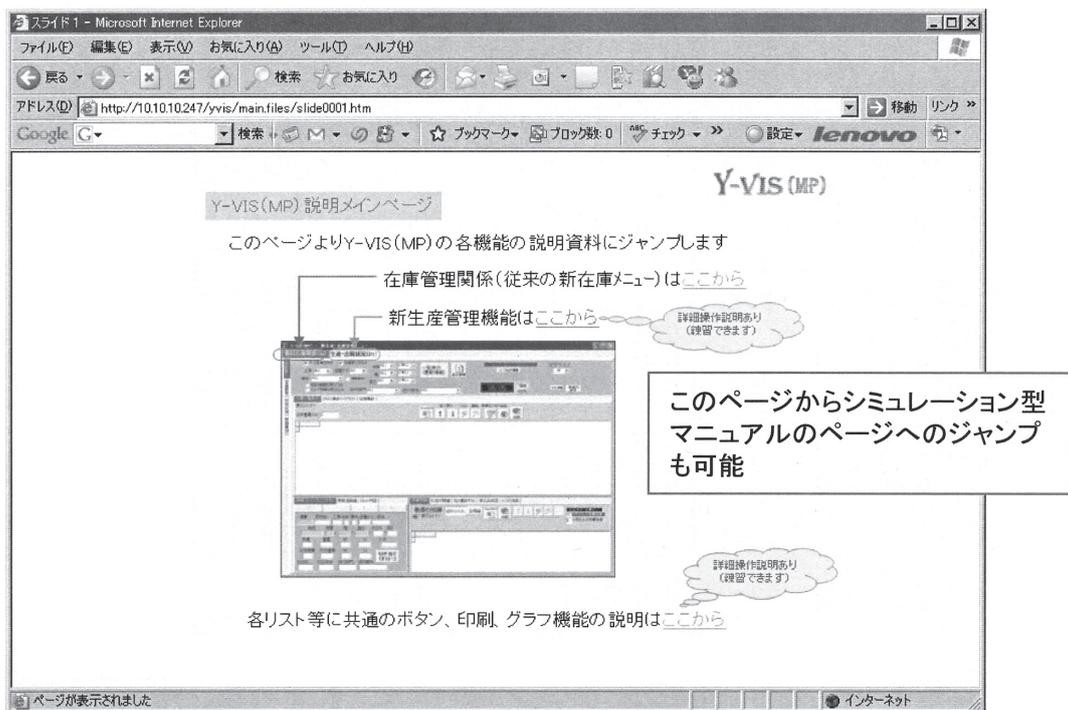


図2-23 共通機能の説明ページ初期画面

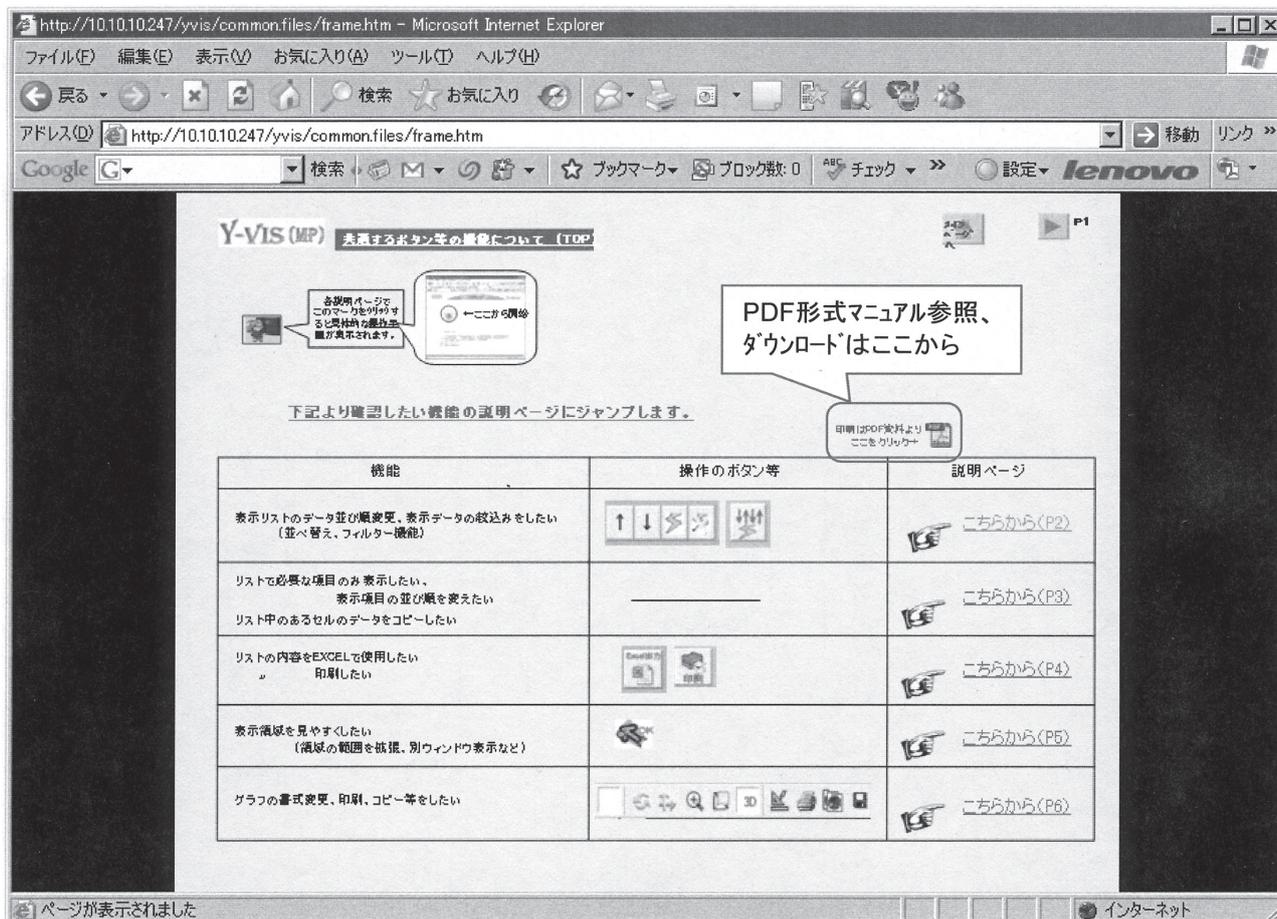


図2-24 リモートデスクトップ機能表示例

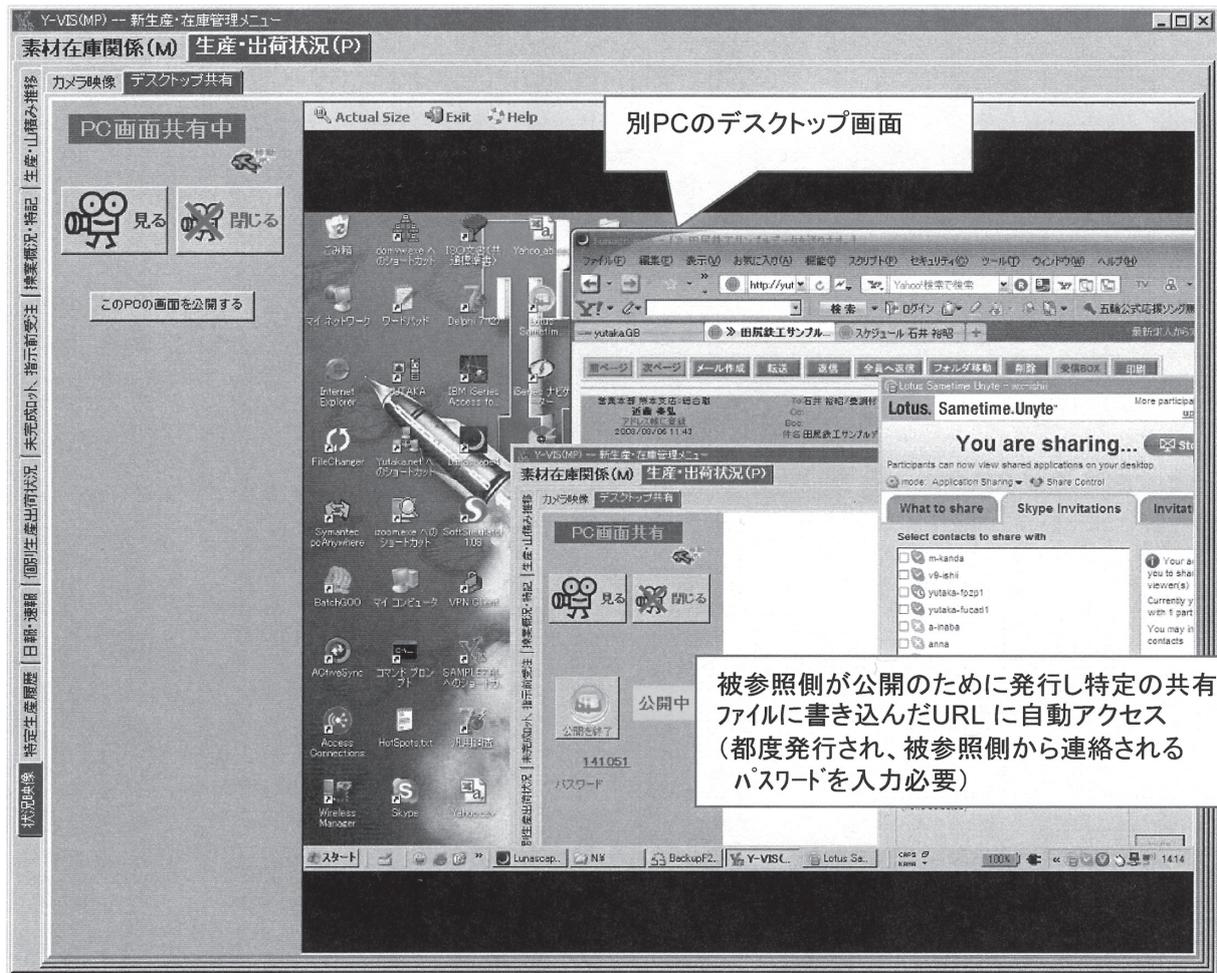


図2-25 お客様住所のGoogleMap表示処理フロー

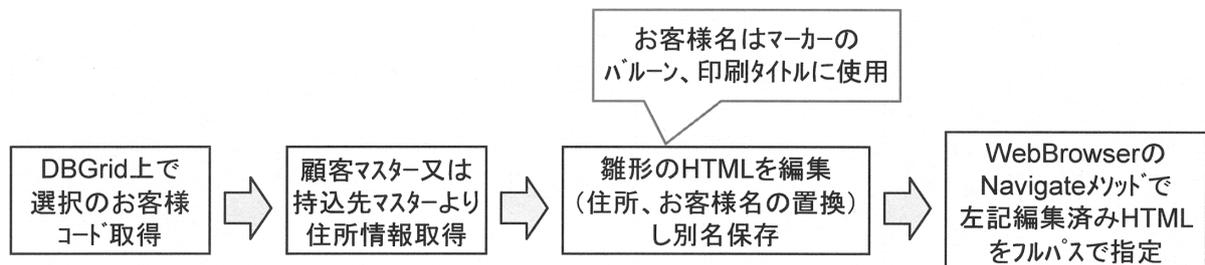


図2-26 お客様住所を表示するHTML記述部分(雛形)

```

if (geocoder) {
  geocoder.getLatLng(
    "KYAKUADD",
    function(point) {
      if (!point) {
        alert("KYAKUADD" + " では表示不能、下のボックスの住所を少し短くして試して下さい(又は古い漢字修正)!");
      } else {
        map.setCenter(point, 12);
        map2.setCenter(point, 15);
        var marker = new GMarker(point);
        var marker2 = new GMarker(point);
        map.addOverlay(marker);
        map2.addOverlay(marker2);
        marker.openInfoWindowHtml("KYAKUNAME");
        marker2.openInfoWindowHtml("KYAKUNAME");
      }
    }
  );
}

```

KYAKUADD、KYAKUNAMEは
実際の住所、お客様名に置換される

図2-27 お客様住所のGoogleMap表示例

The screenshot displays a web interface for production management. On the left is a table titled '受発注別生産(予定含む)・出荷実績' (Production/Shipping Record by Order). The table has columns for '予定' (Forecast), '受発注' (Order), '行' (Line), '客先' (Customer), '持込先' (Destination), '次工程' (Next Process), and '品種' (Product Type). The table lists various orders with their respective details.

The main area shows a Google Map of the Fukuoka region. A callout box indicates '印刷も可能' (Printing is also possible). Another callout box explains '同じ住所を2段階の縮尺で同時表示' (Simultaneous display of the same address at two different zoom levels), showing a zoomed-in view of a specific location labeled 'スチール(株)' (Steel Co.). The interface includes search filters, a date range (2008/07/23 ~ 2008/08/06), and various navigation and printing controls.

図2-28 WebBrowserの印刷処理記述

```

procedure TForm1.MapPrintBTNClick(Sender: TObject);
var
  tmp:OleVariant;
begin
  Form1.WebBrowserMap.ExecWB(OLECMDID_PRINT, OLECMDEXECOPT_DODEFAULT ,tmp, tmp);
end;
    
```

図2-29 map表示不能時の住所内容編集、再試行手段

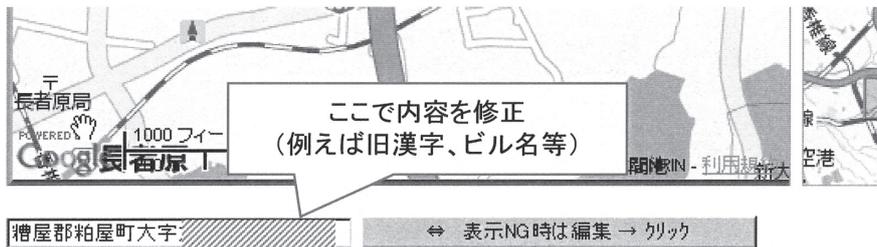


図2-30 ネットワークカメラ表示に関する管理イメージ

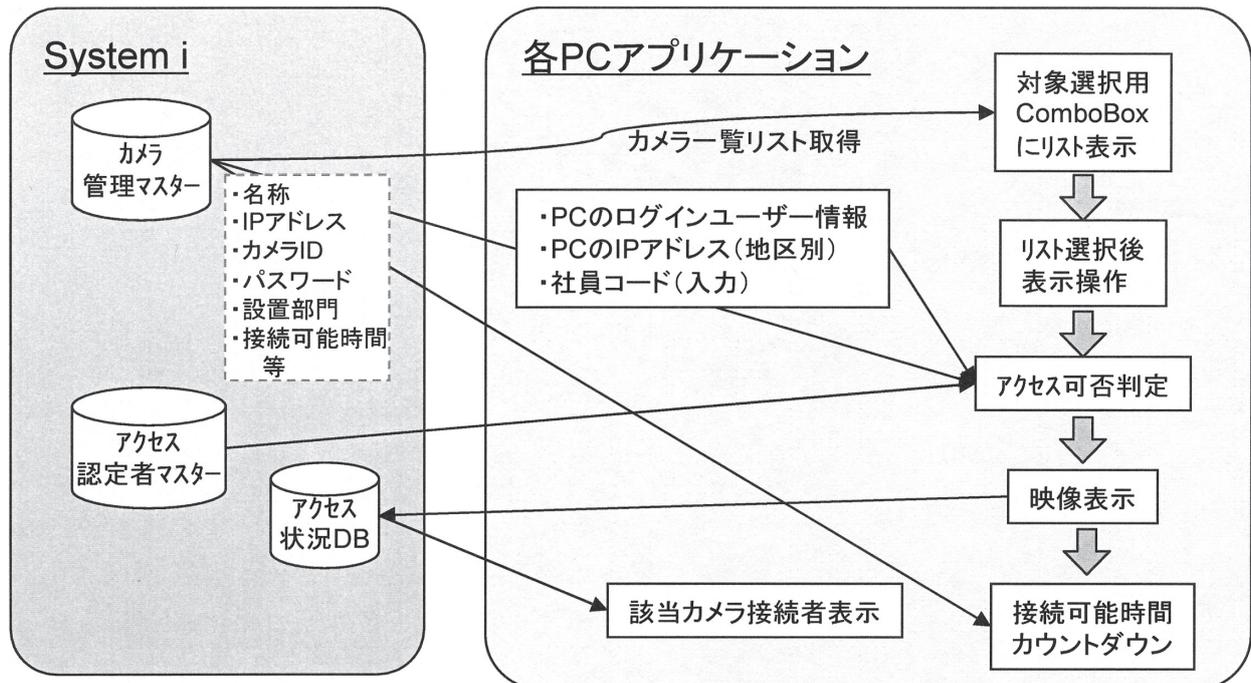


図2-31 ネットワークカメラ表示画面例



図2-32 DecisionGrid (Graph)の集計項目設定フォーム例

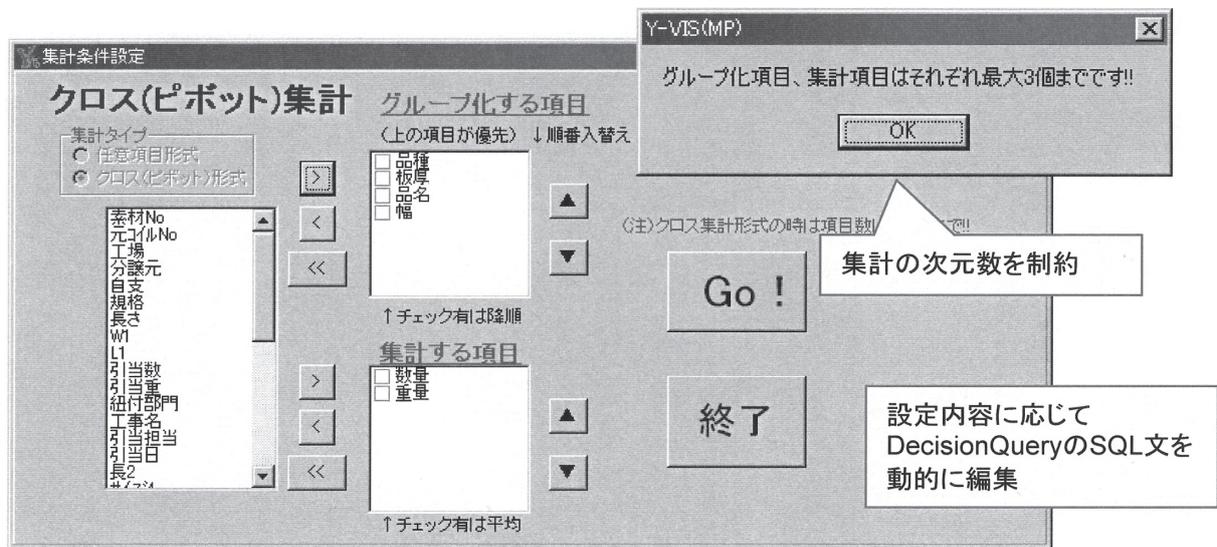


図2-33 DecisionGrid (Graph)の使用例

