

Resizing forms

By Jean-Yves Fock-Hoon, Technical Support Engineer, 4D Inc.

TN 06-16

Overview

このテクニカルノートでは、バージョン **2004** の新しいフォームリサイズコマンドの動作および使用方法について説明しています。

A major change

The problem

この種の問題は、言葉で説明するよりも実際に動いているものを例に出したほうが理解しやすいと思います。本テクニカルノートには、バージョン **6.8** 用のデータベースが付属しています。バージョン **6.8** で開き、カスタムモードで **6.8 Version** メニューの **Demonstration** を選択してみてください。プロセスが **2** 個起動します。第 **1** プロセスでは、スクロールエリアが配置されたウインドウが表示され、スクロールエリアの項目を選択して **Edit** ボタンをクリックすると、第 **2** プロセスのウインドウが表示されるようになっています。それぞれのプロセスは、リストフォームと詳細フォームをイメージしており、一方のフォームをリサイズしてキャンセルすると、他方のフォームに新しいサイズが継承されるような仕組みになっています。この動作は **GET WINDOW RECT** および **SET WINDOW RECT** コマンドを使用して制御しています。

もうひとつは、バージョン **6.8** 用のデータベースを **2004** に変換したデータベースです。バージョン **2004** で開き、カスタムモードで **6.8 Version** メニューの **Demonstration** を選択すると、先程とは動作が違うことに気づきます。リサイズされたウインドウをキャンセルすると、ウインドウサイズは継承されますが、オブジェクトの位置は変わらないため、場合によってはオブジェクトが表示領域外に出てしまう不具合が確認できると思います。

The cause

バージョン **2004** ではコマンドの仕様が変わり、新しいコマンドが追加されるとともに「プッシャー(以降のオブジェクトを移動)」と呼ばれるシステムが導入されました。また、**GET/SET WINDOW RECT** コマンドの動作にも変更が加えられました。バージョン **6.8** から変換されたデータベースが以前とは異なる動きをみせたのはそのためです。

バージョン **6.8** 時代の動作を再現するにはどうすればよいのでしょうか。ドキュメントには、**GET/SET WINDOW RECT** コマンドではなく、**RESIZE FORM WINDOW** コマンドを使用する必要があると記されています。

バージョン 2003 以前では、SET WINDOW RECT コマンドでウインドウをリサイズすると、ウインドウ内に配置されたオブジェクトも自動的にリサイズされました。したがって基本的な設定はできましたが、細かいコントロール、つまりオブジェクトを動かさずにウインドウだけをリサイズするといったことができませんでした。バージョン 2004 ではそのようなこともできます。SET WINDOW RECT コマンドは、ウインドウ内のオブジェクトには作用しなくなり、ウインドウサイズだけが変更されるようになったからです。

The example

2004 Version メニューの Demenstration を選択すると、バージョン 6.8 の動作に戻ります。コードに加えられた変更点は次のようなものです。

6.8 では、単純にウインドウの座標を取得し、インタープロセス変数を通して相手プロセスに渡していました。あとのことは自動リサイズがやってくれるからです。

2004 では、多少のコードを追加する必要があります。

```
GET WINDOW RECT(<>LeftIn2004;<>TopIn2004;<>RightIn2004;<>BottomIn2004)
RESIZE FORM WINDOW(<>RightOut2004-<>LeftOut2004-<>RightIn2004+<>LeftIn2004;<>BottomOut2004<>TopOut2004-<>BottomIn2004+<>TopIn2004)
GET WINDOW RECT($NL1;$NT1;$NR1;$NB1)
SET WINDOW RECT(<>LeftOut2004;<>TopOut2004;$NR1-$NL1+<>LeftOut2004;$NB1-$NT1+<>TopOut2004)
SET WINDOW RECT(<>LeftOut2004;<>TopOut2004;<>RightOut2004;<>BottomOut2004)
```

両方のウインドウ座標を取得していることに注目してください。RESIZE FORM WINDOW でウインドウをリサイズするには、ウインドウサイズの差分が必要だからです。

3 行目以降は最適化できますが、説明のために残されています。RESIZE FORM WINDOW でウインドウサイズの更新はできましたが、ウインドウ位置を継承するためには、同じ座標を渡して SET WINDOW RECT コマンドを発行する必要があります。そのため、上記のコードではウインドウリサイズ後に GET WINDOW RECT コマンドで新しい座標を調べています。これにインタープロセス変数から取得した相手ウインドウ座標との差分を計算して SET WINDOW RECT コマンドを実行すれば同じ位置に同じサイズのウインドウが表示されるはずです。しかし、実のところ RESIZE FORM WINDOW でウインドウサイズはすでに同じになっているので、相手ウインドウ座標をそのまま渡すだけでも結果は変わりません。

したがって実際のコードは次のようなもので充分です。

```
GET WINDOW RECT(<>LeftIn2004;<>TopIn2004;<>RightIn2004;<>BottomIn2004)
RESIZE FORM WINDOW(<>RightOut2004-<>LeftOut2004-<>RightIn2004+<>LeftIn2004;<>BottomOut2004<>TopOut2004-<>BottomIn2004+<>TopIn2004)
SET WINDOW RECT(<>LeftOut2004;<>TopOut2004;<>RightOut2004;<>BottomOut2004)
```

以上がバージョン 6.8 における次のコードに対応しています：

SET WINDOW RECT(<>LeftOut68;<>TopOut68;<>RightOut68;<>BottomOut68)

Resizing or not resizing

4D 2004 では、ウインドウだけをダイレクトにリサイズできるようになりました。以前のバージョンでは、ウインドウをリサイズするとウインドウ内のオブジェクトも一緒にリサイズされました。しかし、考えてみれば、これは別個の機能が単一のコマンドに統合されているようなものでした。リサイズには、ユーザがリサイズハンドルを操作するのと同じ動作をコマンドで実行する場合、あるいは表示領域外にあるオブジェクトを表示するためにウインドウサイズだけを変更する場合の 2 種類があるはずです。

Resizing a window

SET WINDOW RECT はウインドウの座標だけを再定義するコマンドです。とりわけ、表示領域外に配置されたオブジェクトが含まれるようにウインドウをリサイズする場合に便利です。

Example 1:

サンプルの 2004 Version メニューから **Example 1** を選択すると、座標の固定されたリサイズオプションなしのオブジェクトが 3 個配置されたフォームが表示されます。ボタンをクリックするとフォームがリサイズされ、フォームサイズによってオブジェクトが見えたり隠れたりするようになっています。ここまでは以前のバージョンでも実現できるテクニックです。

次に **Example 2** を見てみましょう。リサイズプロパティが拡大に設定されているスクロールエリアの右には隠れたオブジェクトがありますが、いくらウインドウをリサイズしてもスクロールエリアが拡大するだけで表示領域外に配置されたオブジェクトが現れることはありません。これが「以降のオブジェクトを移動する(プッシャー)」オプションです。プッシャーはオブジェクトの右側に配置されたオブジェクトに対して作用します。以前のバージョンでこのようなフォームを作成した場合、ウインドウリサイズによって隠れたオブジェクトが露出してしまいました。フォームの **Advanced** ボタンをクリックすると、ウインドウサイズが現在のサイズ+140 ポイントに拡大され、隠れたオブジェクトが表示されるようになります。この状態でウインドウをリサイズしたとしても、現れたオブジェクトが隠れたりすることはありません。

Example 2:

Example 2 のフォームプロパティを見てみると、フォームサイズは「サイズを決めるもの=Button1」、「制約付き(最大幅=500、最小幅=180)」が設定されていることが分かります。500 に深い意味はありませんが、180 は Button1 がリサイズに伴って **Advanced** ボタンに重ならない最少フォームサイズから算出しています。仮にフォームを 180 ポイントまで縮小して

から **Advanced** ボタンをクリックした場合、隠し領域が加わってフォームサイズが **320** になりますが、この状態でさらにフォームを小さくリサイズすれば、やはり **Button1** が隣のボタンに重なってしまうことになります。**Advanced** ボタンのオブジェクトメソッドで **SET FORM HORIZONTAL RESIZING** を使用しているのはこのためです。このコマンドを使用することにより、隠し領域が表示されている場合とそうでない場合におけるフォームの最大幅と最小幅を設定することができます。

Example 3:

基本的には **Example 2** と同じサンプルですが、違いはプロパティ上のフォームサイズが固定されているにも関わらず、同じように隠し領域が含まれている点です。

これは **On Load** フォームイベントで **SET WINDOW RECT** コマンドを実行していることによります。

Example 4:

Example 4 は「制約付き」オプションの用途を説明するために用意しました。最初に表示されるダイアログは、垂直方向には最大 **300** ポイントまでフォームを伸ばすことができますが、水平方向にはリサイズできません。

Advanced ボタンをクリックすると、スクロールエリアのサイズに基づいてフォームが拡大されます。水平方向にはオフセット(**145**)、垂直方向にはマージン(**10**)が追加されたウインドウにリサイズされる結果、右側に隠された領域が表示されるようになります。さらにこの状態では水平、垂直両方向に最大 **500** ポイントまでウインドウがリサイズできます。ウインドウを最大まで拡大し、**Default** ボタンをクリックして再び元のサイズに戻ろうとすると、オブジェクトメソッドでは次のような処理が行なわれます。はじめに **SET WINDOW RECT** でスクロールエリアのサイズに合わせてウインドウをリサイズします。しかしスクロールエリアは **SET FORM HORIZONTAL RESIZING** および **SET FORM VERTICAL RESIZING** による制約を超えているため、このままでリサイズが実行されません。最後に **RESIZE FORM WINDOW(0;0)** を実行しているにはそのためです。**RESIZE FORM WINDOW(0;0)** を実行すると、ウインドウが再描画され、制約を超えているウインドウは制約が許す範囲内にリサイズされます。

Example 5:

SET WINDOW RECT は、非常に強力なコマンドなので、注意して使用しないとリサイズオプションの設定されたオブジェクトに作用してインタフェースを乱す恐れがあります。**Example 5** のダイアログに配置されたリサイズボタンは、フォームを手動でリサイズしない限りは正常に動作しますが、一旦フォームを手動でリサイズしてしまうと、レイアウトが崩れてしまい、手動でフォームをリサイズしても元の状態には戻れなくなってしまいます。この場合、復帰する

には **MOVE OBJECT** コマンドでオブジェクトを元に戻すしかありません。**Reset** ボタンをクリックしてみてください。

このような状態に陥らないように、オブジェクトプロパティの設定が互いに衝突していないことを確認するようにしてください。パラメータを再定義する場合には、特に注意が必要です。フォームメニュー（またはコンテキストメニュー）から「表示/境界」を選択しておけば、実際に表示されるフォームサイズが常に確認できるので便利です。サイズを決めるものは、デフォルトの設定で「自動サイズ」なので、通常はすべてのオブジェクトが含まれるようなフォームサイズになっているはずです。特定のオブジェクトをサイズを決めるものに設定、あるいは単純にフォームサイズを直接入力すれば、フォームは自動サイズではなくなります。フォームがリサイズできるのであれば、配置されたオブジェクトに拡大または移動のリサイズオプションを設定することができます。フォームの境界外にあるオブジェクトは、リサイズによって他のオブジェクトに干渉することがないように注意を払ってください。リサイズできないフォームであれば、拡大または移動のリサイズオプションを設定するべきではありません。忘れてはならないのは、フォームの最大幅、最小幅、最大高さ、最小高さです。デフォルトの値(0-32000)が残っていると、インタフェースに深刻な問題が生じるかもしれません。

Example 6:

このフォームには、拡大および移動のリサイズオプションが設定されたオブジェクトが配置されています。リサイズオプションなしの四角もあります。**Normal** ボタンをクリックすると、**Open form window** で新しいダイアログが表示されます。**Open form window** はフォームサイズをそのままウインドウサイズとして使用するコマンドです。このとき開かれたウインドウでは、すべてのオブジェクトがフォームエディタでデザインされたとおりにきちんと表示されます。しかし **Extended** または **Shrunk** ボタンをクリックすると、フォームサイズとは異なるサイズのウインドウが開かれるため、オブジェクトの位置関係が激しく崩れてしまいます。特にフォームサイズよりも小さなウインドウを開く **Shrunk** ボタンでは、一部のオブジェクトが縮小されるだけでなく、スタティックテキスト「**Select an item**」は表示もされないという問題が生じます。これはリサイズプロパティが「移動」であるため、小さなウインドウに合わせるとオブジェクトが左に動き、領域の外に出てしまうためです。実際、フォームをリサイズするとオブジェクトがウインドウの中に戻ってきます。

Resizing a window

RESIZE FORM WINDOW は、マウスを操作してウインドウを手動でリサイズするのと動作をするコマンドです。

Example 7:

ダイアログには、移動のリサイズオプションが設定されたオブジェクトが配置されており、ウ

インドウのリサイズに合わせてオブジェクトが移動します。フォームには最大最小サイズも設定されており、その範囲内であれば、値を入力してコマンドでウインドウをリサイズすることもできます。

リサイズにある一定の上限を設けるコマンドは、**SET FORM HORIZONTAL RESIZING** および **SET FORM VERTICAL RESIZING** です。

リサイズを許可する場合、第 1 パラメータに **True** を渡します。第 2、第 3 パラメータはそれぞれ最大最小サイズであり、省略した場合はフォームエディタで設定された値が適用されます。

リサイズを一時的に禁止する場合、第 1 パラメータに **False** を渡します。

いずれのコマンドも、フォームエディタで固定のサイズが設定されているフォームには使用できません。したがってリサイズの有無をコマンドで指定したいフォームには、固定サイズを設定しないでください。どうしても固定サイズが必要なのであれば、コマンドで最大最小サイズを両方ともその値に指定することができます。

例えば、フォームを一定の幅で表示し、手動では水平方向にリサイズできないようにする場合、同じ値の最大幅と最小幅を渡して **SET FORM HORIZONTAL RESIZING** を実行すれば、フォームは幅の固定された状態で表示され、なおかつ任意で **SET FORM HORIZONTAL RESIZING** が実行できるようになります。

水平および垂直リサイズが許可されたフォームは、手動およびコマンドのどちらでもリサイズすることができます。では実際に **RESIZE FORM WINDOW** コマンドを使用してみましょう。コマンドを使用した場合、果たしてどこまでリサイズすることができるのでしょうか。ボタン **Maximum Size** をクリックすると、次のメソッドが実行されます：

RESIZE FORM WINDOW(1000;1000)

コマンドではウインドウサイズに **1000** ポイントを追加するように命じていますが、実際には **600** ポイントという制約に縛られるので、手動でリサイズする場合と同様、**600** ポイントを超えるウインドウサイズに設定することはできません。

同じことはウインドウを縮小する場合にも当てはまります。**Minimum Size** ボタンをクリックすると、今度はウインドウサイズを **1000** ポイント小さくするようなコードが実行されますが、実際に適用されるのは制約による最小サイズです。

フォームの最大幅、最小幅、最大高さ、最小高さをデフォルト値のままにすべきではないと述べたのはこのためです。正しく設定しない場合、ウインドウが使用不可能なサイズになったり、最悪の場合、復帰できなくなるかもしれません。

Note: フォームの最小幅(または最小高さ)を 10 ポイントに設定しても、100 ポイントよりは小さくリサイズできないような場合、フォームにリサイズオプションが「なし」または「拡大」のオブジェクトが置かれていることが原因かもしれません。実際のウインドウ最小幅(および最小高さ)は、そのような移動できないオブジェクトの最小幅(および最小高さ)に依存しています。例えば、サンプルのフォームは最小幅が 200 に設定されていますが、グループボックスおよびボタンのリサイズオプションが「移動」ではないため、実際にはそこまで小さくすることができません。すべてのオブジェクトのリサイズオプションを「移動」に変更すれば、200x200 まではフォームをリサイズすることができます。

Moving a window

SET WINDOW RECT はウインドウを移動するために使用されるコマンドです。GET WINDOW RECT を実行すれば、現在のウインドウ座標が分かるので、それに値を加算または減算して新しい座標を SETWINDOW RECT コマンドに渡せば、ウインドウを移動することができます。

Summary

このテクニカルノートでは、ウインドウ内に配置されたオブジェクトのリサイズ(拡大または移動)を伴うようなウインドウリサイズを実行する方法、およびウインドウサイズだけをリサイズする方法について説明しました。