

クイックレポートエディタ徹底解剖 Part 1

By Kent Wilbur, Manager Information Systems, 4D, Inc.
Technical Note 05-13

(原題: 4D Quick Report Editor Source Code - Part 1)

概要

ビルトインのクイックレポートエディタの正体は、プラグインエリアを含む4Dのフォームである。このTech Noteは、フォーム部分のソースコードを全3回にわたって解説し、流用できそうなもの、デベロッパが変更したいと思われるものを紹介している。

Part 1はフォームの初期設定に関するコードを扱い、Part 2は手動モード、Part 3はウィザードのコードを取りあげている。今回は階層リストのコードが主な内容になるので、クイックレポートとは関係なく、一般的な階層リストのプログラムについても参考になると思う。

クイックレポートのフォームを解析するに際して、各オブジェクトが5つのビューに分類されていることに注意していただきたい。

ビュー1 -アクティブオブジェクト1

ビュー2 -アクティブオブジェクト2

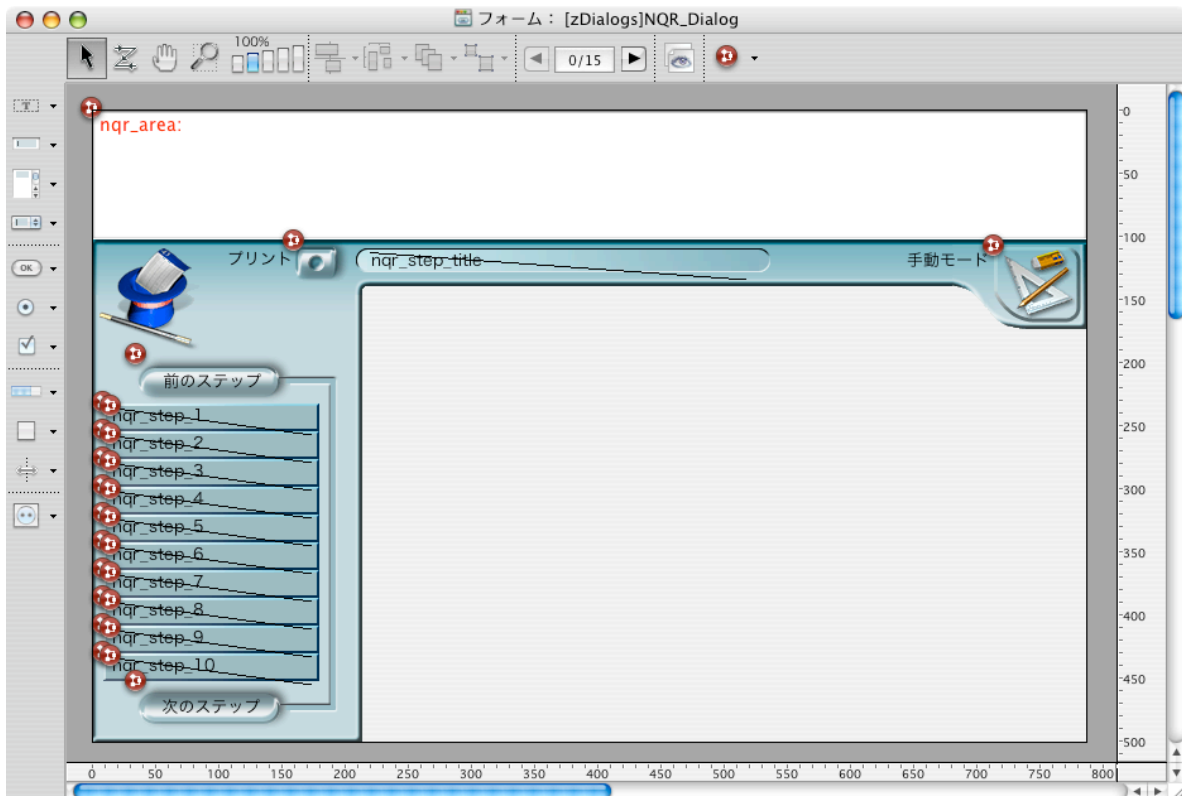
ビュー3 -ボタン

ビュー4 -スタティックオブジェクト

ビュー5 -外観のみのオブジェクト (ピクチャ、背景など。)

クイックレポートのフォームについて

ユーザが普段、目にするクイックレポートエディタは次のような外観をしている。



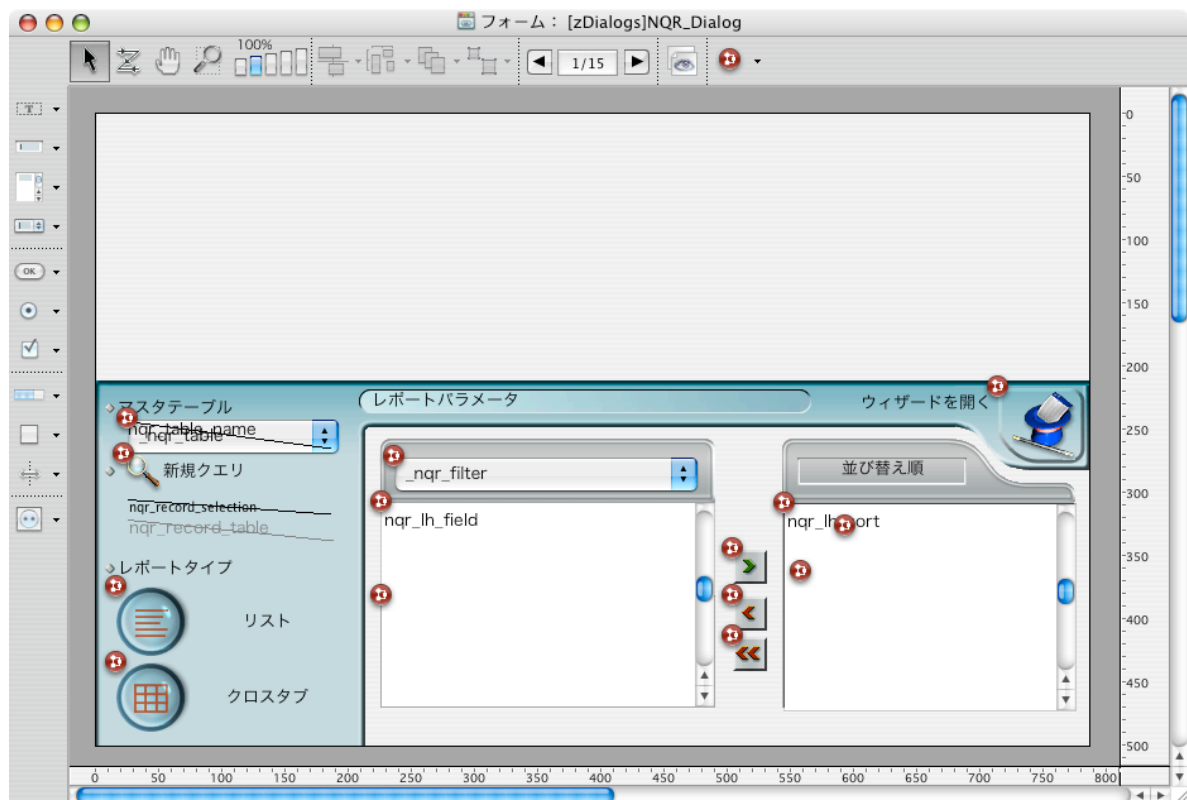
このうち、上の半分はプラグインエリアであり、変更することはできない。唯一、ツールバーオブジェクトの表示/非表示だけがコマンドQR SET AREA PROPERTYによって制御可能である。

デモDBでは、メソッド_Open_dialogによってクイックレポートフォーム "NQR_Dialog"による自作クイックレポートを起動している。変数nqr_wizard,、 nqr_allowsearches,、 nqr_allowautomaticは、それぞれ本来のコマンドQR Reportのブール引数3つに対応している。

```
C_LONGINT($ref) nqr_wizard:=1 `allow Quick Report Wizard
nqr_allowsearches:=1 `allow changing table and queries
nqr_allowautomatic:=1 `allow automatic relations
$ref:=Open form window([zDialogs];"NQR_Dialog")
DIALOG([zDialogs];"NQR_Dialog")
CLOSE WINDOW
```

1ページ目について

1ページ目と継承された0ページ目はこのような外観をしている。



バーチャルストラクチャの構築

クイックレポートエディタが正しく動作するためには、ストラクチャの情報が不可欠である。そのために、バーチャルストラクチャをメモリ上に、つまり複数の1次元/2次元配列にロードする。必要な情報は、テーブル名、フィールド名、テーブル番号、フィールド番号、フィールドの場合は、リレートフィールドなのか、リレートフィールドの場合は、リレートにおける親、子、あるいは親&子、そして自動リレートなのかの情報である。このすべてはメソッドNQR_Init_Struct_Descriptionが処理している。

```
C_LONGINT($id_table;$field;$table_dest;$field_dest;$choicefield)
C_LONGINT($table)
C_LONGINT($nb_table;$id_table)
C_BOOLEAN($autoOne;$autoMany)  ` the arrays for tables
ARRAY TEXT(_nqr_table;0)
ARRAY LONGINT(_nqr_table_id;0)
GET TABLE TITLES(_nqr_table;_nqr_table_id)  ` the two dimension arrays
for fields by table
$nb_table:=Count tables
ARRAY TEXT(_nqr_field;$nb_table;0)
ARRAY LONGINT(_nqr_field_id;$nb_table;0)  ` the arrays for handling the
relationships
ARRAY INTEGER(_nqr_table_child;0)
ARRAY INTEGER(_nqr_field_child;0)
ARRAY INTEGER(_nqr_table_parent;0)
ARRAY INTEGER(_nqr_field_parent;0)
ARRAY BOOLEAN(_nqr_Oneauto;0)
ARRAY BOOLEAN(_nqr_manyauto;0)  ` loop on the list of the VISIBLE
tables to find the fields and the relations

For ($table;1;Size of array(_nqr_table_id);1)  ` get the number of the
table to work upon
$id_table:=_nqr_table_id{$table}  ` read the list of the fields and place
them in the tables created for this purpose
GET FIELD TITLES(Table($id_table)->,$_field;$_id)
COPY ARRAY($_field;_nqr_field{$id_table})
```

```

COPY ARRAY($_id;_nqr_field_id{$id_table})      ` research the relations to
the current table
For ($field;1;Count fields($id_table);1)      ` read the nature of the
relationship
GET RELATION
PROPERTIES($id_table;$field;$table_dest;$field_dest;$choicefield;$autoOne;$autoMany)      ` if a relationship exists
If ($field_dest#0) & ($table_dest#0)          ` read the properties of the
relationship in the context of the current table
GET FIELD RELATION(Field($id_table;$field)->,$one2;$many2;*)
Case of
\ ($one2=2)
$autoOne:=False
\ ($one2=3)
$autoOne:=True
Else ` defensive code (useless in theory)
$autoOne:=False
End case
Case of
\ ($many2=2)
$autoMany:=False
\ ($many2=3)
$autoMany:=True
Else ` defensive code (useless in theory)
$autoOne:=False
End case      ` add the values to the relationship arrays
APPEND TO ARRAY(_nqr_table_child;$id_table)
APPEND TO ARRAY(_nqr_field_child;$field)
APPEND TO ARRAY(_nqr_table_parent;$table_dest)
APPEND TO ARRAY(_nqr_field_parent;$field_dest)
APPEND TO ARRAY(_nqr_Oneauto;$autoOne)
APPEND TO ARRAY(_nqr_manyauto;$autoMany)
End if
End for
End for ` sort the table arrays by alphabetical order
SORT ARRAY(_nqr_table;_nqr_table_id)

```

余談だが、このメソッドはクイックレポートに限らず非常に便利である。ストラクチャ情報がいつでも参照できることの意味は大きい。一時的に特定のリレートだけを自動リレートにしたいとき、すべてのリレートを自動リレートにするコマンド SET AUTOMATIC RELATIONS(True;True) では問題がある。ストラクチャ情報を参照すれば、あるリレートについて、それが自動かどうかを調べることができ、2004の新コマンド SET FIELD RELATION を使えば一時的に自動リレートにすることができる。（これだけで別のTech Note にする価値があると思う。）

例題:

```
SET AUTOMATIC RELATIONS(False;False) ` Set the relations to their
default settings
NQR_Init_Struct_Description
$LTable:=Table(Current default table)    ` Loop through the relationship
arrays
For ($i;1;Size of array(_nqr_table_parent))    ` Is the current table a
parent table in a relationship
If (_nqr_table_parent{$i}=$LTable)    ` Is the relation currently set to a
manual relation
If (Not(_nqr_manyauto{$i}))    ` Make the relation automatic
SET FIELD
RELATION(Field(_nqr_table_child{$i};_nqr_field_child{$i})->;Automatic ;Aut
omatic )
End if
End if
` Is the current table a child table in a relationship
If (_nqr_table_child{$i}=$LTable)    ` Is the relation currently set to
a manual relation
If (Not (_nqr_Oneauto{$i}))    ` Make the relation automatic
SET FIELD
RELATION(Field(_nqr_table_child{$i};_nqr_field_child{$i})->;Automatic ;Aut
omatic )
End if
End if
End for
```

フィールド名の階層リストを作成する

フォームのOn loadイベント、またポップアップ（マスターテーブル;リレートテーブル;すべてのテーブル）が更新されるたびにメソッドNQR_MP_Get_Fieldsがコールされて、使用できるフィールド名とリレートテーブルの階層リストを構築している。

```
`Initialization of variables
C_INTEGER($table;$field)
C_TEXT($title)
C_BOOLEAN($index)
C_LONGINT($repeat;$length;$hide;$size;$type;$style)
C_TEXT($format) C_LONGINT(nqr_lh_field)
C_LONGINT($i;$find;$tt;$ttt;$ref_field2;$element)
C_LONGINT($field_l;$find_table;$find_field;$subfield;$ref_subfield;$ref_field;$sub_list;$ref_element)
C_LONGINT($subfield_list)
C_LONGINT($table_parent;$ref_field3;$table_child)
C_POINTER($fieldptr)
C_TEXT($text) `initialization of the possible dialogs of subfields
ARRAY LONGINT($subfield_id;0)
```

はじめに階層リストをクリアするわけだが、ここでは同じ階層リストを使用するページ数だけ用意している。同じ階層リストを同一フォームの複数ページ、あるいは同一プロセスの複数フォームで使用することはできない。階層リストは、展開される可能性があり、表示状態が場所によって異なるとプログラムが破綻するためである。それで階層リストは、使用する場所の数だけ用意する。

```
If (Is a list(nqr_lh_field))
CLEAR LIST(nqr_lh_field;*)
End if
If (Is a list(nqr_lh_field2))
CLEAR LIST(nqr_lh_field2;*)
End if
If (Is a list(nqr_lh_field3))
CLEAR LIST(nqr_lh_field3;*)
End if `defensive code is case the number of the current table is lost ...
```

```

If (_nqr_table=0) | (_nqr_table>Size of array(_nqr_table))
$table:=QR
Get report table(nqr_area)
$find:=Find in array(_nqr_table_id;$table)
If ($find>0)
_nqr_table:=$find
Else
_nqr_table:=1
End if
End if

```

次に、セクション中のレコード数、テーブル内の全レコード数を表示するためのテキストを取得している。ここではリソース14907を参照しているので表示される文字列は各言語の4Dによって異なる。これもまたフォームに手を加えずにテキストを変更するためのテクニックである。

```

`assigning the information about the table
nqr_record_selection:=String(Records in
selection(Table(_nqr_table_id[_nqr_table])>))+" "+Get indexed
string(14907;18) `record(s) in selection"
nqr_record_table:=String(Records in
table(Table(_nqr_table_id[_nqr_table])>))+" "+Get indexed
string(14907;19) `record(s) in table"
nqr_table_name:=_nqr_table[_nqr_table]

```

次にリストを表示用に初期化する。

```

`create the list
nqr_lh_field:=New list
SET LIST PROPERTIES(nqr_lh_field;0;0;18;1)

```

表示されるリストを特定するために3つの変数を使用されている。カレントテーブル=nqr_cb_1、リレートテーブル=nqr_cb_2、すべてのテーブル=nqr_cb_3である。次のコードは1と2で使用されるものである。


```

Case of
\ (nqr_cb_1=1) | (nqr_cb_2=1) `list the fields of the current table report or
related tables report
`get the number of the current table
$table:=_nqr_table_id[_nqr_table]          `loop on the fields of the current
table
For ($field_1;1;Size of array(_nqr_field{$table});1)          `get the field
number
$field:=_nqr_field_id{$table}{$field_1}          `get the field type
$type:=Type(Field($table;$field)->)          ` get the position of the field in
the table of the identifiers
$find_table:=Find in array(_nqr_table_id;$table)
If ($find_table>0)
$find_field:=Find in array(_nqr_field_id{$table};$field)
Else `defensive code in case of a problem
$find_field:=-1
End if

```

次のフィールドタイプの例外を処理する。BLOBタイプはクイックレポートにできない。サブテーブルは、仮の階層リストを作成し、もとのリストに追加 (APPEND) する。

```

Case of
\ ($find_table<0) `problem with the table; nothing to make
\ ($find_field<0) `problem with the field; nothing to make
\ ($type=ls BLOB ) `is a blob ; nothing to make because a blob can't be in
a quick report
\ ($type=ls Subtable ) `is a subtable ; build a list of the subfields
`read the list of subfields
GET FIELD TITLES(Field($table;$field)->,$_subfield,$_subfield_id) `not
supported, but functional
$_subfield_list:=New list
For ($_subfield;1;Size of array($_subfield);1)          `build a reference
to the subfield
$ref_subfield:=($_subfield_id{$_subfield} << 24)+($table << 16)+$field

```

```
APPEND TO LIST($subfield_list;$_subfield{$subfield};$ref_subfield)
SET LIST ITEM PROPERTIES($subfield_list;$ref_subfield;False;Plain ;
1150+Type(Field($table;$field;$_subfield_id{$subfield})->))
End for
```

4Dのプログラミングでは、本来フィールド名は必要がない。テキストのフィールド名を要求するコマンドは存在しないのであり、必要なのは、テーブル番号とフィールド番号mあるいはフィールドに対するポインタである。一方、ユーザは番号ではなくフィールド名を期待している。階層リストの各アイテムにはユニークな参照番号（倍調整数）があるので、フィールド番号はそれで管理するとして、ではテーブル番号はどうすればいいだろうか。それにフィールドがサブテーブルの場合、サブフィールドは同じフィールド番号を共有していながら、サブフィールド内では独自のユニークなフィールド番号を持っていなければならない。ここで、ひとつの倍長整数の中に3つの倍調整数情報を収める手法が必要になる。ビットマッピングである。

4Dには最大で255テーブルが存在し得る。これはつまり、2バイト、あるいは8ビットの情報である。テーブルには512フィールド、これは3バイトあるいは12ビットである。残りはサブテーブル用に使用すればよい。

```
$ref_subfield:=($_subfield_id{$subfield} << 24)+($table << 16)+$field
```

これで必要な情報がすべてひとつの倍長整数になった。わかりやすくするために、分解して考えてみよう。

```
$ref_subfield = "0x00020002"  Table 2  Field 2
$ref_subfield = "0x00020003"  Table 2  Field 3
$ref_subfield = "0x02030001"  Subfield 2 Table 3  Field 1
```

最初の1バイト（0xの次の2桁）は、サブフィールド番号である。次の1バイトは、テーブル番号、次がフィールド番号である。これで、階層リストのアイテム参照番号に、すべての情報を収めることができる。

```

`build a reference to the field
$ref_field:=( $table << 16)+$field          `add the field to the list
APPEND TO
LIST(nqr_lh_field;_nqr_field{$table}{$find_field};$ref_field;$subfield_list;True)
SET LIST ITEM
PROPERTIES(nqr_lh_field;$ref_field;False;Plain ;1150+Type(Field($table;$field)->))

```

次は、サブテーブル以外の普通のフィールドを階層リストに加えるものである。さらに、インデックス属性がある場合は、フォントを太字に設定している。

```

Else `is a valid field; add this one to the list          `build a reference
to the field
$ref_field:=( $table << 16)+$field          `see if the field is indexed
GET FIELD PROPERTIES($table;$field;$type;$length;$index)
If ($index)
$style:=Bold
Else
$style:=Plain
End if

```

次のコードはリレートフィールドおよびリレート先のテーブルに関する処理である。階層リストには、リレートテーブルごとにサブリストが必要である。リレートに関する情報は、サブリストの参照番号に収める。つまり、リレートの有無、自動か否か、などである。フィールド用の参照番号にはすでに3つの情報を割り当てているので、これらの情報の管理には別フォーマットでサブリストの参照番号を使用する。リレート情報であることは、冒頭の1ビットを目印として使用する。なお、冒頭の1ビットを1にするということは、その値は負の値であるということである。

繰り返しになるが、4Dにはテーブルが255までしか存在しないので、その特定には8ビットだけで十分である。倍調整数の残り8ビットでリレート情報を記録するわけだが、問題は、リレート関係における親なのか子なのかという点である。そこで、上位4ビットに情報がある場合は子テーブル、下位4ビットにある場合は親テーブルというルールを定めることにする。

```
`the selected reference is - the number of the shifted relation of 16 bits  
$ref_field2:=-($tt << 16)
```

これで必要な6種類の情報を倍長整数である階層リストの参照番号に収めることができた。まとめると次のとおりである。

```
$ref_subfield HEX = "0xSSTTFFFF" (The main list reference number  
contents)  
$ref_subfield HEX = "0xFFFFFTT" (Related table is a parent table  
format)  
$ref_subfield HEX = "0xFFTT0000" (Related table is a child table  
format)
```

最後に階層リストの処理を行なう。If (\$sub_List=0) というのはIf(Is a list(\$sub_list)) というのと同じ意味で、0の場合は階層リストが存在しない。

```
If (nqr_cb_1=1) ` If we are not showing related tables we are finished  
`add the field to the list  
APPEND TO LIST(nqr_lh_field;nqr_field{$table}{$find_field};$ref_field)  
SET LIST ITEM  
PROPERTIES(nqr_lh_field;$ref_field;False;$style;1150+Type(Field($table;$  
field)->))  
Else ` check for related items  
$sub_list:=0 `set sublist flag to zero `look on the known  
relations  
For ($tt;1;Size of array(_nqr_table_child);1) `is the  
relationship outbound or a child relationship  
If (_nqr_table_child{$tt}=$table) & (_nqr_field_child{$tt}=$field) `this is the  
starting field of a child relationship  
If (_nqr_Oneauto{$tt}) `the relationship is automatic (the calculation of the  
report will be possible) `find the parent table  
$find_table:=Find in array(_nqr_table_id;nqr_table_parent{$tt})  
If ($find_table>0) `if the sublist does not exist yet,  
to create it  
If ($sub_list=0)  
$sub_list:=New list
```

```

End if                                     `the selected reference is - the number of
the relation                             $ref_field2:=-$tt
`add the table to the list
APPEND TO LIST($sub_list;"["+_nqr_table{$find_table}+"]";$ref_field2)
SET LIST ITEM PROPERTIES($sub_list;$ref_field2;False;Bold ;14930)
End if
End if
End if                                     `is the relationship a return or parent relationship
If (_nqr_table_parent{$tt}=$table) & (_nqr_field_parent{$tt}=$field) `this is
the related field in the parent table
If (_nqr_manyauto{$tt}) & (nqr_allowautomatic=1) `the relationship is
automatic (the calculation of the report will be possible) and the
relationship returns are authorized
`find the child table
$find_table:=Find in array(_nqr_table_id;_nqr_table_child{$tt})
If ($find_table>0)                       `if the sublist does not exist yet,
to create it
If ($sub_list=0)
$sub_list:=New list
End if                                     `the selected reference is - the number of
the shifted relation of 16 bits
$ref_field2:=-($tt << 16)                `add the table to the list
APPEND TO LIST($sub_list;"["+_nqr_table{$find_table}+"]";$ref_field2)
SET LIST ITEM PROPERTIES($sub_list;$ref_field2;False;Bold ;14931)
End if
End if
End if
End for                                     `add the field to the list with the possibility of related
tables
APPEND TO
LIST(nqr_lh_field;_nqr_field{$table}{$find_field};$ref_field;$sub_list;True)
SET LIST ITEM
PROPERTIES(nqr_lh_field;$ref_field;False;$style;1150+Type(Field($table;$
field)->))
End if
End case
End for

```

これで必要な作業は済んだようにみえるかもしれないが、クイックレポートエディタは数テーブル離れたリレートもサポートしているので、親の親、あるいは子の子に関する処理を、リレート関係が途絶えるまでたどる必要がある。

```
If (nqr_cb_2=1) `list related tables
$element:=0 `one starts at the beginning !
`analyse all of the elements of the obtained list
While ($element<Count list items(nqr_lh_field))
$element:=$element+1 `pass to the element according to
`information on the element and in particular when an associated list is
attached
GET LIST ITEM(nqr_lh_field;$element;$ref_element;$text;$sub_list)
If ($ref_element<0) `if the reference is negative then it can have a cascade
If (((-$ref_element) >> 16)>0) `a relationship exists
`to see the parent table
$table_parent:=_nqr_table_child{((- $ref_element) >> 16)}
`loop on all the relations
For ($ttt;1;Size of array(_nqr_table_parent))
If (_nqr_table_parent{$ttt}=$table_parent) `this is the parent table
If (_nqr_manyauto{$ttt}) & (nqr_allowautomatic=1) `the relationship is
automatic (the calculation of the report will be possible) and the
relationship returns are authorized
`find the child table in the array
$find_table:=Find in array(_nqr_table_id;_nqr_table_child{$ttt})
If ($find_table>0) `if the sublist does not exist yet,
to create it
If ($sub_list=0)
$sub_list:=New list
End if
$ref_field3:=-( $ttt << 16)
APPEND TO LIST($sub_list;"["+_nqr_table{$find_table}+"]";$ref_field3)
SET LIST ITEM PROPERTIES($sub_list;$ref_field3;False;Bold ;14931)
End if
End if
End if
```

```

End for                                `modify the element to add the new sublist under
this element
SET LIST
ITEM(nqr_lh_field;$ref_element;$text;$ref_element;$sub_list;True)
Else                                  `is a relationship from the child table
`find the parent table
$table_child:=_nqr_table_parent{- $ref_element}                                `loop on all
the relations
For ($ttt;1;Size of array(_nqr_table_child))
If (_nqr_table_child{$ttt}=$table_child) `This is the child table
If (_nqr_Oneauto{$ttt}) `the relationship is automatic (the calculation of
the report will be possible)
`find the parent table in the array
$find_table:=Find in array(_nqr_table_id;_nqr_table_parent{$ttt})
If ($find_table>0)                    `if the sublist does not exist yet,
to create it
If ($sub_list=0)
$sub_list:=New list
End if
$ref_field3:=-$ttt
APPEND TO LIST($sub_list;"["+_nqr_table{$find_table}+"]";$ref_field3)
SET LIST ITEM PROPERTIES($sub_list;$ref_field3;False;Bold ;14930)
End if
End if
End if
End for                                `modify the element to add the new sublist under
this element
SET LIST
ITEM(nqr_lh_field;$ref_element;$text;$ref_element;$sub_list;True)
End if
End if
End while
End if

```

すべてのテーブルを表示するのはたやすい。リレートなど無視して、非表示のフィールドを除くすべてのフィールドリストのアイテムとして加え、それぞれをサブリストとしてもとのリストに追加するだけである。

```

\ (nqr_cb_3=1) `list of all the tables          `loop throught all the tables
For ($i;1;Size of array(_nqr_table_id);1)      `get the table number
$table:=_nqr_table_id{$i}                      `create a new list
$sub_list:=New list                           `loop on the fields of the table
For ($field_l;1;Size of array(_nqr_field{$table});1) `get the field
number
$field:=_nqr_field_id{$table}{$field_l}        `get the field type
$type:=Type(Field($table;$field)->)           ` get the position of the field
in the table of the identifiers
$find_table:=Find in array(_nqr_table_id;$table)
If ($find_table>0)
$find_field:=Find in array(_nqr_field_id{$table};$field)
Else `defensive code in case of a problem
$find_field:=-1
End if
Case of
\ ($find_table<0) `problem with the table; nothing to make
\ ($find_field<0) `problem with the field; nothing to make
\ ($type=ls BLOB ) `is a blob ; nothing to make because a blob can't be in
a quick report
\ ($type=ls Subtable ) `is a subtable ; build a list of the subfields
`read the list of subfields
GET FIELD TITLES(Field($table;$field)->,$_subfield;$_subfield_id)
$subfield_list:=New list
For ($subfield;1;Size of array($_subfield);1)
$ref_subfield:=($_subfield_id{$subfield} << 24)+($table << 16)+$field
APPEND TO LIST($subfield_list;$_subfield{$subfield};$ref_subfield)
SET LIST ITEM PROPERTIES($subfield_list;$ref_subfield;False;Plain ;
1 150+Type(Field($table;$field;$_subfield_id{$subfield})->))
End for
$ref_field:=($table << 16)+$field
APPEND TO
LIST($sub_list;_nqr_field{$table}{$find_field};$ref_field;$subfield_list;True)
SET LIST ITEM
PROPERTIES($sub_list;$ref_field;False;Plain ;1 150+Type(Field($table;$fiel
d)->))

```



```

Else
$ref_field:=( $table << 16)+$field
APPEND TO LIST($sub_list;_nqr_field{$table}{$find_field};$ref_field)
GET FIELD PROPERTIES($table;$field;$type;$length;$index)
If ($index)
$style:=Bold
Else
$style:=Plain
End if
SET LIST ITEM
PROPERTIES($sub_list;$ref_field;False;$style;1 150+Type(Field($table;$field)->))
End case
End for
APPEND TO LIST(nqr_lh_field;"["+_nqr_table{$i}+"]";$table << 16;$sub_list;False)
SET LIST ITEM PROPERTIES(nqr_lh_field;$table << 16;False;Bold ;450)
End for

```

クイックレポートエディタのある箇所では、カラムを表示するために階層リストが使用されている。参考までにそのコードは以下のとおりである。

```

\ (nqr_cb_5=1) `list columns
For ($i;1;QR Count columns(nqr_area))
QR GET INFO
COLUMN(nqr_area;$i;$title;$fieldptr;$hide;$size;$repeat;$format)
APPEND TO LIST(nqr_lh_field;$title;$i)
SET LIST ITEM PROPERTIES(nqr_lh_field;$i;False;Plain ;14932)
End for
End case

```

あとはリストのコピーを作成し、各階層リストをそれぞれ更新するだけである。

```

`copy the hierarchial lists
nqr_lh_field2:=Copy list(nqr_lh_field)
nqr_lh_field3:=Copy list(nqr_lh_field)

```

`update the new list properties

SET LIST PROPERTIES(nqr_lh_field2;0;0;18;1)

SET LIST PROPERTIES(nqr_lh_field3;0;0;18;1)

`redraw the lists in the dialog

REDRAW LIST(nqr_lh_field)

REDRAW LIST(nqr_lh_field2)

REDRAW LIST(nqr_lh_field3)

`posting of the fields (in the event of relations)

NQR_MP_View_Fields

`posting of the sorting

NQR_MP_Get_Sorts

コールされているのは次のメソッドNQR_MP_View_Fieldsである。

`initialization

C_BOOLEAN(\$index;\$continue)

C_LONGINT(\$i;\$field_l;\$find_table;\$find_field;\$subfield)

C_LONGINT(\$length)

C_LONGINT(\$list_field;\$list_detail;\$h1;\$h2;\$h3)

C_LONGINT(\$item;\$ref_field;\$table;\$field;\$type;\$style)

C_LONGINT(\$subfield_list;\$ref_subfield)

C_TEXT(\$suffix;\$text)

このメソッドは配置場所からリストを特定しているので、もし別の場所に階層リストを配置したのであれば、変更を忘れないようにする必要がある。

`choose the co-ordinates according to the page

\$continue:=True

Case of

\ (Current form page=1)

\$list_field:=nqr_lh_field

\$list_detail:=nqr_lh_field_detail

\$h1:=162

\$h2:=84

```

$h3:=90
$suffix:=""
\ (Current form page=3)
$list_field:=nqr_lh_field2
$list_detail:=nqr_lh_field_detail2
$h1:=158
$h2:=64
$h3:=70
$suffix:="2"
\ (Current form page=7)
$list_field:=nqr_lh_field3
$list_detail:=nqr_lh_field_detail3
$h1:=205
$h2:=110
$h3:=116
$suffix:="3"
Else `defensive code
$continue:=False
End case

```

選択されたアイテムがフィールドかサブフィールドである場合は、通常の処理でリストを動かしている。

```

If ($continue) `get the information on the selected item
$item:=Selected list items($list_field)
GET LIST ITEM($list_field;$item;$ref_field;$text)
Case of
\ ($ref_field>=0) `is a field or subfield
`get the position of the main list
GET OBJECT RECT(*;"nqr_lh_field"+$suffix);$g;$h;$d;$b)
`move the main list
MOVE OBJECT(*;"nqr_lh_field"+$suffix);$g;$h;$d;$h+$h1;*)
`the secondary list is rejected if it is out of the visible zone
MOVE
OBJECT(*;"nqr_lh_field_detail"+$suffix);10000;10000;10000;10000;*)
SELECT LIST ITEMS BY POSITION($list_field;$item)

```

選択されたアイテムがリレートテーブル（つまり負の値）である場合、元のフィールドリストはフォーム上の別の場所に移動され、代わりにリレートテーブルのフィールドリストが表示される。リストの作成自体は前述のコードと同じである。

```
\ ($ref_field<0) `is a table
`get the position of the main list
GET OBJECT RECT(*;"nqr_lh_field"+$suffix);$g;$h;$d;$b)
`move the main list
MOVE OBJECT(*;"nqr_lh_field"+$suffix);$g;$h;$d;$h+$h2;*)
`the other half is for the secondary list
MOVE OBJECT(*;"nqr_lh_field_detail"+$suffix);$g;$h+$h3;$d;$h+$h1;*)
SELECT LIST ITEMS BY REFERENCE($list_field;$ref_field)

`initialize the list of the dependent fields
If (Is a list($list_detail))
CLEAR LIST($list_detail;*)
End if
$list_detail:=New list
SET LIST PROPERTIES($list_detail;0;0;18;1)

`get the table number of the dependent table
If (((-$ref_field & 0x00FFFFFF) >> 16)>0)
$table:=$_nqr_table_child{(-$ref_field & 0x00FFFFFF) >> 16}
Else
$table:=$_nqr_table_parent{(-$ref_field & 0x00FFFFFF)}
End if

`loop on the fields of the dependent table
For ($field_l;1;Size of array(_nqr_field{$table});1)
$field:=_nqr_field_id{$table}{$field_l}
$type:=Type(Field($table;$field)->)
$find_table:=Find in array(_nqr_table_id;$table)
If ($find_table>0)
$find_field:=Find in array(_nqr_field_id{$table};$field)
Else
$find_field:=-1
End if
```

```

Case of
\ ($find_table<0)
\ ($find_field<0)
\ ($type=ls BLOB )
\ ($type=ls Subtable )
GET FIELD TITLES(Field($table;$field)->$_subfield;$_subfield_id)
$subfield_list:=New list
For ($subfield;1;Size of array($_subfield);1)
$ref_subfield:=( $_subfield_id{$subfield} << 24)+($table << 16)+$field
APPEND TO LIST($subfield_list;$_subfield{$subfield};$ref_subfield)
SET LIST ITEM PROPERTIES($subfield_list;$ref_subfield;False;Plain ;
1150+Type(Field($table;$field;$_subfield_id{$subfield})->))
End for

$ref_field:=( $table << 16)+$field
APPEND TO
LIST($list_detail;_nqr_field{$table}{$find_field};$ref_field;$subfield_list;True)
SET LIST ITEM
PROPERTIES($list_detail;$ref_field;False;Plain ;1150+Type(Field($table;$field)->))

Else
$ref_field:=( $table << 16)+$field
GET FIELD PROPERTIES($table;$field;$type;$length;$index)
If ($index)
$style:=Bold
Else
$style:=Plain
End if
APPEND TO
LIST($list_detail;"["+_nqr_table{$find_table}+"]"+_nqr_field{$table}{$find_field};$ref_field)
SET LIST ITEM
PROPERTIES($list_detail;$ref_field;False;$style;1150+Type(Field($table;$field)->))
End case

```

```
End for
SELECT LIST ITEMS BY POSITION($list_detail;1)
```

最後はリストのリドローである。

```
`update the information in the lists
Case of
\ (Current form page=1)
nqr_lh_field_detail:=$list_detail
GOTO AREA(nqr_lh_field_detail)
REDRAW LIST(nqr_lh_field)
If (Is a list(nqr_lh_field_detail))
SET LIST PROPERTIES(nqr_lh_field_detail;0;0;18;1)
REDRAW LIST(nqr_lh_field_detail)
End if
\ (Current form page=3)
nqr_lh_field_detail2:=$list_detail
GOTO AREA(nqr_lh_field_detail2)
REDRAW LIST(nqr_lh_field2)
If (Is a list(nqr_lh_field_detail2))
SET LIST PROPERTIES(nqr_lh_field_detail2;0;0;18;1)
REDRAW LIST(nqr_lh_field_detail2)
End if
\ (Current form page=7)
nqr_lh_field_detail3:=$list_detail
GOTO AREA(nqr_lh_field_detail3)
REDRAW LIST(nqr_lh_field3)
If (Is a list(nqr_lh_field_detail3))
SET LIST PROPERTIES(nqr_lh_field_detail3;0;0;18;1)
REDRAW LIST(nqr_lh_field_detail3)
End if
End case
End case
End if
```

その他の処理

フォームをユーザに解放するまでにしなければならない処理は他にもある。多くの場合、On loadイベントに限らず随時コールされるものである。

ウィザード使用、新規クエリなどの初期設定によるもの。

```
`set the wizard available options
SET VISIBLE(*;"wizard@";(nqr_wizard=1))
If (nqr_allowsearches=1)
SET VISIBLE(*;"table@";True)
SET VISIBLE(nqr_table_name;False)
Else
SET VISIBLE(*;"table@";False)
SET VISIBLE(nqr_table_name;True)
End if
```

レポートタイプによるもの。

```
`set up the types of reports : (1) list report (2) cross tab report
$type_report:=QR Get report kind(nqr_area)
If ($type_report=1)
nqr_tr_1:=1    nqr_tr_2:=0
SET FORMAT(nqr_tr_1;"1;4;;14934;64")
SET FORMAT(nqr_tr_2;"1;4;;14935;208")
Else
nqr_tr_1:=0
nqr_tr_2:=1
SET FORMAT(nqr_tr_2;"1;4;;14935;64")
SET FORMAT(nqr_tr_1;"1;4;;14934;208")
DISABLE BUTTON(*;"bp1 @")
NQR_MP_Get_tc_sort
End if
```

“1;4;14934;64”のような記述に慣れていないならば、この機会に覚えておくとう便利である。なにかといえば、フォーム上の画像をプログラムの制御するための手法で、イメージをピクチャライブラリ、ピクチャ変数（データからロードしても良い）、PICTリソースに保存しておき、状況に合わせて画像を変更するためのテクニックである。この例でいうと、1列、4行の画像、ピクチャリソース#14934を参照、そして64は背景を透過するという意味である。特に2004ではユーザ変更フォームが導入された結果、安易にフォームエディタでフォームを修正できなくなったので、この手法は非常に有効であると思う。

メソッドNQR_MP_Get_tc_sortおよびNQR_MP_Get_Sortについては、Part 2で取り上げる予定である。

まとめ

クイックレポートエディタの解説と銘打ちながら、汎用的なプログラミングテクニックの話になってしまった。メソッドNQR_Init_Struct_Descriptionは、バーチャルストラクチャの構築として幅広く応用できるし、メソッドNQR_MP_Get_Fieldsは階層リストの優れた例題である。また、ビットマッピングを使用して、複数のデータを効率的に管理する方法についても取り上げた。クイックレポートエディタの解説はPart 3まであるので、続編も是非、参照して欲しい。