



この文書では、4th Dimension および 4D Server バージョン 6.8.1 の新機能について、以下の項目別に説明しています。

- MacOS X 特有の機能の使用方法
- 新しい印刷処理コマンド
- その他の新しいコマンド（割込み、オブジェクトプロパティ、フォームイベントおよび 4D 環境テーマ）
- IMAP4 および SSL の 4D インターネットコマンドへの統合（この新機能については、別のドキュメントで詳しく説明されています）

MacOS X 特有の機能の使用方法

4th Dimension バージョン 6.8.1 では、グラフィック・レンダリング・エンジンサポート (Quartz) およびパルシング・デフォルトボタンの使用により、MacOS X プラットホーム・インタフェース特有の機能をより良く処理することや、MacOS X および Windows XP のプラットフォームをスタイルシートに追加できます。

グラフィック・レンダリング・エンジンサポート

MacOS X プラットホームには、Quartz というグラフィック・レンダリング・エンジンが備えられています。それにより、アプリケーションは本質的にはアンチエイリアスの技術に基づき、テキストや画像をよりきれいに表示することができます。この技術は“スムージング”とも呼ばれ下図のように文字の縁を滑らかにするものです。

標準表示



アンチエイリアス処理された表示



4th Dimension バージョン 6.8.1 では、コード変更を行わずに 4D フォームでこの技術を使用することができます。

注：Quartz エンジンは表示のみに使用されます。印刷は標準モードで行われます。

関係するオブジェクト

以下の通り、4Dフォームオブジェクトのほとんどにアンチエイリアス処理の表示を反映させることができます。

- スタティック・テキストエリア
- 標準ボタンのタイトル
- チェックボックス
- ラジオボタン
- タブコントロール
- ポップアップ／ドロップダウンリスト
- 階層ポップアップメニュー
- コンボボックス（リストのみ）
- サーモメータ／ルーラオブジェクト
- グループボックス
- スクロールエリア
- 階層リスト

レンダリングのアクティブ化

MacOS Xでは、Quartz アピアランスマネージャは常にアクティブになっています。フォームにアンチエイリアス効果を反映させるには、以下の条件を満たしている必要があります。

- データベースを MacOS X プラットホーム上で実行している
- 「Mac テーマ」 プラットホーム・インタフェースが使用されている
- 文字フォントサイズが、システムオプション “滑らかな文字を使用しないフォントサイズ…以下” (MacOS X システム環境設定の一般コントロールパネル) で指定されるサイズより大きい

上記の条件を満たしていない場合は、標準のグラフィック・レンダリング (MacOS 9 タイプ) が使用されます。

テキストの長さ

アンチエイリアスを使用すると、テキストサイズがわずかに大きくなります。オブジェクトのサイズがテキストを完全に表示するのに不十分な場合は、テキストは切り詰められ省略符号が付きます。例として、“Cancel and quit” というテキストは “Cancel and qu...” と表示されます。

これにより省略されたテキストを容易に識別することができます。しかしボタン、チェックボックスおよびラジオボタン等のオブジェクトタイトルには、複数行のテキストが含まれる場合があります。このような場合はオブジェクトの高さがあれば長すぎるテキストは次の行へ移動されます。テキストにキャリッジリターン (Char(13)) が含まれる場合は、キャリッジリターン以下の部分が次の行へ移動されます。

パルシング・デフォルトボタン

4th Dimension バージョン 6.8.1 では、フォームにパルシングボタンを使用することができます。パルシングボタンとは、MacOS X では下図のようなダイアログボックスのデフォルトボタンを指します。



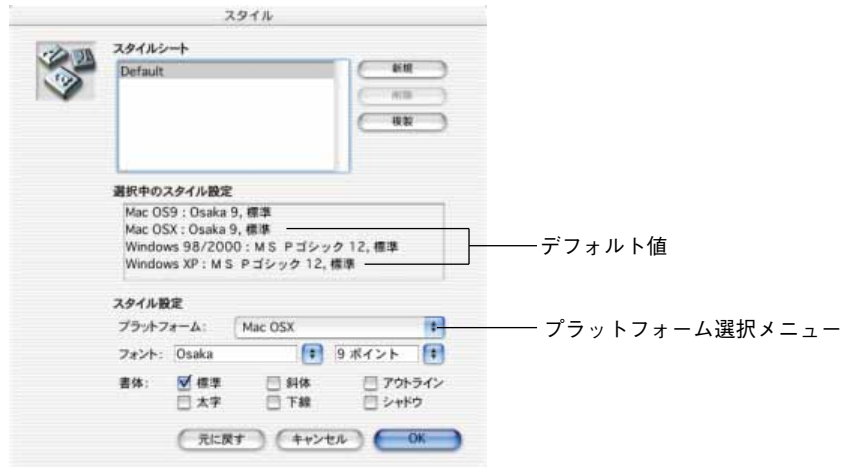
フォームにパルシングボタンを表示させるには、以下の条件を満たしている必要があります。

- データベースを MacOS X プラットホーム上で実行している
- 「Mac テーマ」 プラットホーム・インタフェースが使用されている
- ボタンがデフォルトボタンである

新しいプラットフォームに対応したスタイルシート

Mac OSX および Windows XP の 2 つの新しいプラットフォームに、個別のスタイルシートを指定できるようになりました。

この 2 つのプラットフォームは、図のようにスタイルシートの「定義」ダイアログボックスで選択できます。



フォントおよびフォントサイズを直接入力できるようになりました（コンボボックスを使用）。これによりインストールされていないフォント、およびフォントサイズを指定できるようになります。

スタイルシートの互換性

4th Dimension バージョン 6.8.1 を使用して作成または変更したスタイルシートは、バージョン 6.8 との互換性を持ちません（バージョン 6.8 で開いた際にデータベースに変更を加えることができません）。また、データベースをバージョン 6.8.1 で開き、その後バージョン 6.8 で変更を加えると、バージョン 6.8.1 では反映されません。

ルートセッションを開かずに 4D Web サービスを使う

MacOS X では、Web 公開用に確保される TCP/IP ポート（ポート 0～1023）を使用するには特定のアクセス権が必要であり、マシンの“ルート”ユーザのみがこれらのポートを使ってアプリケーションを起動することができます。4D データベースを Web 上に公開するには“ルート”ユーザとして接続するか、またはデータベース公開用の TCP ポート番号を変更する必要があります。

こうした方法では不十分な場合があるため、4D 社では MacOS X 上で 4D Web サービスを公開するための新しい手法を提案します。この方法では“ルート”ユーザになる必要はなく、また 4D Web サービスへ接続する URL に特別なポート番号を付加する必要もありません。

この新しい方法はポート転送に基づくものです。この原理は、システムレベルで標準のTCPポート番号（80）で受け取った要求を4Dデータベースで指定されたポート番号（1023より大きい番号であること）へ転送するというものです。この方法は、SSL接続では用いることができない点に注意してください（TCP443ポートは変更できません）。

この設定を行うには、“ルート”ユーザとして接続しターミナルを起動してUnixコマンドを使用する必要があります。

▼ MacOS Xでポート転送（IPアドレスが192.168.93.45であると仮定）を設定するには、次のように行います。

1. ルートユーザとしてセッションを開く。

この操作については、『4th Dimension 6.8 アップデート』マニュアルで概説されています。

2. Terminalプログラムを起動する。

このプログラムは、/Applications/Utilities/のフォルダで見つけることができます。

3. “su”（“代理ユーザ”の特別なアカウント）と入力し、[return]キーを押す。続けて、ルートユーザのパスワードを入力する。

4. 以下のコマンドを入力する。

```
ipfw add 400 fwd 192.168.93.45,8080 tcp from any to 192.168.93.45 80
```

上記のコマンドを入力する際は、“192.168.93.45”の部分自身のIPアドレスに置き換えてください。400という数字はこの操作の参照番号です。

5. Terminalプログラムを終了する。

6. 標準ユーザとして4Dアプリケーションを起動する。

7. データベースプロパティで、Web公開用TCPポートを8080に設定する。

これにより、Mac OSXはユーザに対して透過的な方法でポート80で受け取った要求をポート8080に即座に転送することが可能になります。

▼ このオペレーティングモードを削除するには、次のように行います。

1. ターミナルプログラムを起動し、以下のコマンドを入力する。

```
ipfw delete 400
```

これにより、ポート80で受け取った要求はポート8080に転送されなくなります。

印刷処理

4th Dimension バージョン 6.8.1 では、印刷処理を目的とする以下のような新しいコマンドがいろいろ充実しています。

- Get print marker
- SET PRINT MARKER
- Get printed height
- GET PRINTABLE MARGIN
- SET PRINTABLE MARGIN
- GET PRINTABLE AREA

さらに、Print form コマンドが関数（値を返す）になり、追加の引数も同様に認められるようになりました。

このようなコマンドはすべて“印刷”テーマにあります。

Get print marker

Get print marker (markNum) → 数値

引数	タイプ		説明
markNum	数値	→	マーカー番号
戻り値	数値	←	マーカーの位置

Get print marker コマンドで印刷の際のマーカーの現在位置を受け取ることができます。

このコマンドは、**PRINT SELECTION** および **PRINT RECORD** コマンドで **On header** フォームイベントの実行時のみ使用できます。

座標はピクセル単位（1ピクセル=1/72インチ）で返されます。

以下の“Form area”テーマの定数のうちの1つを markNum 引数に渡します。

定数	タイプ	値
Form Header	倍長整数	200
Form Header1 ... 10	倍長整数	201...210
Form Detail	倍長整数	0
Form Break0 ... 9	倍長整数	300...309
Form Footer	倍長整数	100

参照：SET PRINT MARKER、MOVE OBJECT

SET PRINT MARKER

SET PRINT MARKER (markNum; position)

引数	タイプ	説明
markNum	数値	→ マーカー番号
position	数値	→ マーカーの新しい位置

SET PRINT MARKER コマンドで、印刷の際のマーカーの位置を指定できます。このコマンドを **Get print marker** および **MOVE OBJECT** コマンドと組み合わせて使用することにより、印刷範囲のサイズを調節できます。

SET PRINT MARKER コマンドは、**PRINT SELECTION** および **PRINT RECORD** コマンドとの関連で、On header フォームイベントにのみ使用できます。コマンドの効果は印刷に限られており、変更は画面上には表示されません。フォームに対して行った変更は保存されません。

以下の“Form area” テーマの定数のうちの1つを markNum 引数に渡します。

定数	タイプ	値
Form Header	倍長整数	200
Form Header1 ... 10	倍長整数	201...210
Form Detail	倍長整数	0
Form Break0 ... 9	倍長整数	300...309
Form Footer	倍長整数	100

マーカーの position はピクセル単位で指定します。

注意：

このコマンドは既存のマーカーの位置のみを変更します。マーカーを追加することはできません。フォームに存在しないマーカーを指定した場合は、コマンドは機能しません。

また、デザインモードで定義したマーカー順序は維持されるため、他のマーカーを飛び越すような移動はできません。

参照：Get print marker、MOVE OBJECT

印刷可能マージンの処理

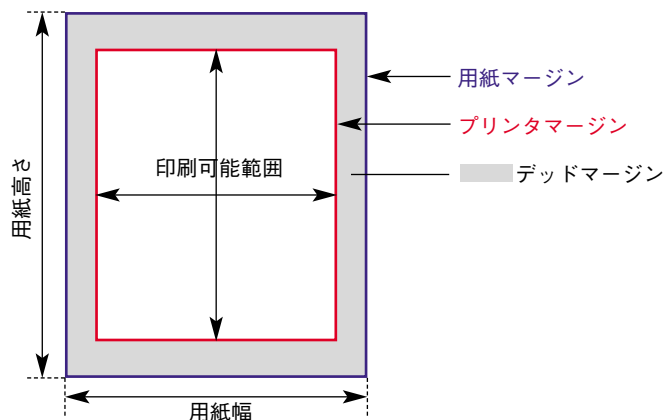
4th Dimension バージョン 6.8.1 では印刷の際のマージンの処理方法が変わりました。

デフォルトでは、4th Dimension における印刷に関わる計算は“印刷可能マージン”に基づいています。このシステムの利点は、フォームが自動的に新しいプリンタに適応するという点です（印刷可能範囲内に配置されているため）。その一方で、事前印刷用紙の場合はプリンタを変更すると印刷可能マージンが変わるため、正確に印刷されるようにエレメントを配置することができませんでした。

4th Dimension バージョン 6.8.1 より、**Print form**、**PRINT RECORD** および **PRINT SELECTION** コマンドを使って行うフォームの印刷を、各プリンタに共通の固定マージン（言い換えると用紙マージン、つまり用紙の物理的な範囲）に基づいて行うことができますようになりました。これは、新しい **GET PRINTABLE MARGIN**、**SET PRINTABLE MARGIN** および **GET PRINTABLE AREA** コマンドを使って行うことができます。

印刷用語

- 用紙マージン：用紙マージンは用紙の物理的な範囲に相当します。
- プリンタマージン：プリンタマージンは、その外側をプリンタが印刷できないことを示すマージンです（プリントローラー、プリンタヘッドのエンドオブトラベル等の物質的な理由により）。プリンタマージンはプリンタやフォーマットごとに異なります。
- デッドマージン：これは用紙マージンとプリンタマージン間の範囲を指します。



GET PRINTABLE MARGIN

GET PRINTABLE MARGIN (left; top; right; bottom)

引数	タイプ	説明
left	数値	← 左マージン
top	数値	← 上マージン
right	数値	← 右マージン
bottom	数値	← 下マージン

GET PRINTABLE MARGIN コマンドは、**Print form** コマンドを使って指定されたそれぞれのマージンの現在の値を返します。

値は、用紙の端からの長さがピクセル単位で返されます。

GET PRINTABLE AREA 関数を使って、用紙サイズを受け取ることや、印刷可能範囲を計算することもできます。

参照：Print form、GET PRINTABLE AREA、SET PRINTABLE MARGIN

SET PRINTABLE MARGIN

SET PRINTABLE MARGIN (left; top; right; bottom)

引数	タイプ	説明
left	数値	→ 左マージン
top	数値	→ 上マージン
right	数値	→ 右マージン
bottom	数値	→ 下マージン

Print form コマンドの使用時に、**SET PRINTABLE MARGIN** コマンドで様々な印刷マージンの値を設定できます。

以下の値のうちの1つを、left、top、right、bottomの引数に渡すことができます。

■ 0 = 用紙マージンを使用

■ -1 = プリンタマージンを使用

■ 値 > 0 = ピクセル単位のマージン (72 dpiでの1ピクセルは、約0.4 mm)

rightおよびbottom引数の値はそれぞれ、用紙の右端および下端からの長さを表します。

デフォルトでは4th Dimensionは、プリンタマージンに基づいて印刷を行います (前述の『印刷可能マージンの処理』の節を参照)。**SET PRINTABLE MARGIN** コマンドを実行すると、変更された引数はセッション全体を通して同じプロセスで保持されます。

▼ 次の例では、デッドマージンのサイズを受け取ることができます。

```
SET PRINTABLE MARGIN (-1;-1;-1;-1) 印刷可能マージンを設定します
```

```
GET PRINTABLE MARGIN($l;$t;$r;$b)
```

`\$l、\$t、\$r および \$b は、用紙のデッドマージンに相当します

▼ 次の例では、用紙サイズを受け取ることができます。

```
SET PRINTABLE MARGIN (0;0;0;0) 用紙マージンを設定します
```

```
GET PRINTABLE AREA($height;$width)
```

`A4 サイズの場合：\$height=842；\$width=595（単位：ピクセル）

参照：Print form、Get printed height、GET PRINTABLE MARGIN

GET PRINTABLE AREA

GET PRINTABLE AREA (height{; width})

引数	タイプ	説明
height	数値	← 印刷可能範囲の高さ
width	数値	← 印刷可能範囲の幅

GET PRINTABLE AREA コマンドは、印刷可能範囲の height および width 引数のサイズをピクセル単位で返します。このサイズは、現在の印刷設定、用紙方向等に左右されます。

返されるサイズは1ページごと（例えば、ページブレイクの後）に異なるものではありません。

Get printed height と関連してこのコマンドは印刷可能なピクセル数の確認やページ上のオブジェクトのセンタリングに有効です。

ページ全体のサイズを確認するには、次のように行います。

■ **GET PRINTABLE MARGIN** コマンドで受け取ったマージンに、このコマンドで返された値を加えます。

■ もしくは以下の構文を使用します。

```
SET PRINTABLE MARGIN(-1;-1;-1;-1) 印刷可能マージンを設定します
```

```
GET PRINTABLE AREA(hPaper;wPaper) 用紙サイズ
```

参照：Print form、GET PRINTABLE MARGIN

Get printed height

Get printed height → 数値

引数	タイプ	説明
----	-----	----

このコマンドには、引数は必要ありません

戻り値	数値 ←	マーカーの位置
-----	------	---------

Get printed height コマンドは、**Print form** コマンドを使って印刷されるセクション全体の高さ（ピクセル単位）を返します。

返される値は、0（ページ上端）から **GET PRINTABLE AREA** コマンドによって返される全体の高さ（印刷可能範囲の最大サイズ）までの間の値です。

Print form コマンドを使って新しいセクションを印刷する場合は、新しいセクションの高さがこの値に加えられます。このセクションに対して使用できる印刷可能範囲が不十分である場合は、新規ページが作成され返される値は0になります。

右および左の印刷可能マージンは、上および下のマージン（**SET PRINTABLE MARGIN** コマンドを使って指定）とは異なり、返される値に影響しません。

参照：Print form、GET PRINTABLE AREA、SET PRINTABLE MARGIN

Print form

Print form ({table;} form{; area1{; area2}) → 数値

引数	タイプ	説明
Table	テーブル	→ 印刷するテーブル、または、省略する場合は、デフォルトテーブル
form	文字列	→ 印刷するフォーム
area1	数値	→ 印刷マーカー、または開始エリア (area2が指定される場合)
area2	数値	→ 終了エリア (area1が指定される場合)
戻り値	数値	← 印刷されるセクションの高さ

4Dバージョン6.8.1より、**Print form** コマンドは2つの追加の引数（オプション）を認め、また値を返すことができるようになりました。

以下の2つの構文を使用できます。

■ フォームエリア印刷

height:=Print form (myTable;myForm;marker)

この場合コマンドは **marker** で示されるセクションを印刷します。以下の “Form area” テーマの定数のうちの1つを **marker** 引数に渡します。

定数	タイプ	値
Form Header	倍長整数	200
Form Header1 ... 10	倍長整数	201...210
Form Detail	倍長整数	0 (デフォルト値)
Form Break0 ... 9	倍長整数	300...309
Form Footer	倍長整数	100

■ セクション印刷

`height:=Print form (myTable;myForm;areaStart;areaEnd)`

この場合、コマンドは `areaStart` および `areaEnd` 引数間のセクションを印刷します。値はピクセルで単位で入力しなければいけません。

Print form コマンドによって返される値は印刷可能範囲の高さを表します。この値は、**Get printed height** コマンドに自動的に考慮されます。

参照：GET PRINTABLE AREA、Get printed height

その他の新しいコマンド

4Dバージョン 6.8.1にはいくつかの新しいコマンドが加わりました。その新機能は以下のような様々なテーマに関わるものです。

■ 割込み： **Method called on error**、**Method called on event**

■ オブジェクトプロパティ： **Get alignment**、**SET ALIGNMENT**

■ フォームイベント： **Right click**、**Contextual click**

■ 4D環境： **Is license available**

割込みテーマ

Method called on error

Method called on error → 文字列

引数	タイプ	説明
このコマンドには、引数は必要ありません		
戻り値	文字列	← エラー時に呼出されるメソッド名

Method called on error コマンドは、**ON ERR CALL** コマンドによってカレントプロセスに組み込まれたメソッド名を返します。

そのようなメソッドが組み込まれていない場合は、空白の文字列 ("") が返されます。

▼ このコマンドを使ってエラーキャッチング・メソッドを一時的に変更し、その後リストアできるため、コンポーネントとの関連で特に有効です。

```
$methCurrent:=Method called on error
ON ERR CALL("NewMethod")
` 文書を開くことができない場合に、エラーが生じます
$ref :=Open document("MyDocument.l")
` 前のメソッドの再インストール
ON ERR CALL($methCurrent)
```

参照：ON ERR CALL

Method called on event

Method called on event → 文字列

引数	タイプ	説明
このコマンドには、引数は必要ありません		
戻り値	文字列	← イベント時に呼出されるメソッド名

Method called on event コマンドは、**ON EVENT CALL** コマンドによって組み込まれたメソッド名を返します。

そのようなメソッドが組み込まれていない場合は、空白の文字列 ("") が返されます。

参照：ON EVENT CALL

オブジェクトプロパティ・テーマ

Get alignment

Get alignment ({*; }object) → 数値

引数	タイプ	説明
*		→ 指定した場合、 object はオブジェクト名 (文字列) 省略した場合、フィールドまたは変数
object	オブジェクト	→ オブジェクト名 (*を指定した場合)、 またはフィールドまたは変数 (*を省略 した場合)
戻り値	文字列	← 文字揃えコード

Get alignment コマンドは、**object** および * 引数によって指定されたオブジェクトに適用される文字揃えのタイプを示すコードを返します。

オプションの * 引数を指定した場合は、**object** 引数にはオブジェクト名 (文字列) を指定します。* 引数を省略した場合は、**object** 引数にはフィールドまたは変数を指定します。この場合、文字列ではなくフィールドまたは変数のリファレンス (フィールドまたは変数オブジェクトのみ) を指定します。

注：コマンドに複数のオブジェクトを指定すると、最後のオブジェクトの文字揃えの値のみが返されます。

返されるコードは “Object alignment” テーマにある以下の定数のいずれかです。

定数	タイプ	値
Align default	倍長整数	1
Align left	倍長整数	2
Center	倍長整数	3
Align right	倍長整数	4

文字揃えが適用されるフォームオブジェクトは、以下の通りです。

- スクロールエリア
- コンボボックス
- スタティック・テキスト
- ポップアップメニュー／ドロップダウンリスト
- フィールド
- 変数

参照：SET ALIGNMENT

SET ALIGNMENT

SET ALIGNMENT ({*; }object; alignment)

引数	タイプ	説明
*		→ 指定した場合、objectはオブジェクト名 (文字列) 省略した場合、フィールドまたは変数
object	オブジェクト	→ オブジェクト名 (*を指定した場合)、 またはフィールドまたは変数 (*を省略 した場合)
alignment	数値	→ 文字揃えコード

SET ALIGNMENT コマンドを使って、objectおよび*引数によって指定されたオブジェクトに適用される文字揃えのタイプを設定できます。

オプションの*引数を指定した場合は、object引数にはオブジェクト名 (文字列) を指定します。*引数を省略した場合は、object引数にはフィールドまたは変数を指定します。この場合、文字列ではなくフィールドまたは変数のリファレンス (フィールドまたは変数オブジェクトのみ) を指定します。

以下の“Object alihnment” テーマの定数のうちの1つをalignment引数に渡します。

定数	タイプ	値
Align default	倍長整数	1
Align left	倍長整数	2
Center	倍長整数	3
Align right	倍長整数	4

アライメントが適用されるフォームオブジェクトは、以下の通りです。

- スクロールエリア
- コンボボックス
- スタティック・テキスト
- グループボックス
- フィールド
- 変数

参照：Get alignment

フォームイベント・テーマ

Right click

Right click → ブール値

引数	タイプ	説明
このコマンドには、引数は必要ありません		
戻り値	ブール値	← 右クリックが検出された場合はTrue、そうでない場合はFalse

Right click コマンドはマウスの右ボタンをクリックした場合にTrueを返します。

このコマンドは **On clicked** フォームイベントで実行されたフォームメソッド／オブジェクトメソッド内でのみ使用します。そのため、デザインモードにおいてフォームプロパティと特定のオブジェクトの両方またはどちらかで、イベントが正しく選択されていることを確認する必要があります。

注：このコマンドはWindowsおよびMacOS Xでのみ動作します。

MacOS 9では常にFalseが返されます。

参照：Form event、Contextual click

Contextual click

Contextual click → ブール値

引数	タイプ	説明
このコマンドには、引数は必要ありません		
戻り値	ブール値	← コンテキストチュアルクリックが検出された場合は True、そうでない場合は False

Contextual click コマンドはコンテキストチュアルクリックが行われた場合に True を返します。

■ Windows ではコンテキストチュアルクリックは、マウスの右ボタンを使って行います。

■ MacOS ではコンテキストチュアルクリックは、Ctrl キーを押しながらクリックして行います。

このコマンドは、**On clicked** フォームイベントで実行されたフォームメソッド／オブジェクトメソッド内でのみ使用します。そのため、デザインモードにおいてフォームプロパティと特定のオブジェクトの両方またはどちらかで、イベントが正しく選択されていることを確認する必要があります。

▼ このメソッドをスクロールエリアと組み合わせることにより、コンテキストメニューを使って配列要素の値を変更できます。

```
If(Contextual click)
  If(Pop up menu("True;False")=1)
    myArray{myArray}:="True"
  Else
    myArray{myArray}:="False"
  End if
End if
```

参照：Form event、Right click

4D 環境テーマ

Is license available

Is license available{(license)}→ブール値

引数	タイプ	説明
license	数値	→ ライセンスの有効性テストを行う製品／プラグイン
戻り値	ブール値	← 製品／プラグインが、使用可能な場合は True、そうでない場合は False

このコマンドを使って、製品やプラグインの利用可能性を確認できます。例として、プラグインが必要な機能を表示または非表示にする際に有効です。

Is license available コマンドは次のような3通りの使用が可能です。

■ license 引数を省略する場合：4D アプリケーションがデモモードの場合に、コマンドは False を返します。

■ 以下の “Is license available” テーマの定数のうちの1つを license 引数にパスする場合。

定数	タイプ	値
4D Draw License	倍長整数	808464694
4D for OCI License	倍長整数	808465208
4D View License	倍長整数	808465207
4D Web License	倍長整数	808464945
4D Write License	倍長整数	808464697

このとき、対応する製品がロードされライセンスが使用可能な場合（4D Server で）に、コマンドは True を返します。

例として、4D Draw のシリアルナンバーがあり expansion ナンバーがない場合は、コマンドは 4th Dimension のシングルステーション・システムでは True を、4D Server では False を返します。反対に、4D Draw の expansion ナンバーがありシリアルナンバーがない場合は、コマンドは 4D Server では True を、4th Dimension では False を返します。シリアルナンバーおよび expansion ナンバーのどちらもある場合は、コマンドはいかなる場合も True を返します。

■ プラグイン “4BNX” リソースの ID 番号を license 引数に直接パスする場合：コマンドは、前述の通りに動作します。