

4D v18.x リリースノート

4D v18 へようこそ。以下は、このリリースでの新機能・変更された機能についての情報をまとめたページへのリンクです。

- [What’s new in 4D v18](#) - 4D v18 の新機能と改善点をまとめたblog 記事です
- [Fixed bugs list for 4D v18.2](#) - 4D v18.2 で修正された全てのバグの一覧です。
- [Fixed bugs list for 4D v18.1](#) - 4D v18.1 で修正された全てのバグの一覧です。
- [Fixed bugs list for 4D v18](#) - 4D v18 で修正された全てのバグの一覧です。.
- [変更点とアップデートについて](#) - 既存の開発に影響を与えうる、新リリースでの変更点の一覧です。
- [廃止予定または削除された機能](#) - カレントのリリースにおいて、**廃止予定**(将来削除される予定)または **削除済み**となった機能、設定、コマンドなどの一覧です。

注:

- お使いのデータベースを4D v18 へと変換することを検討しているのであれば、専用の [4D v18への変換](#) のマニュアルもご覧ください。
- バイナリーデータベースをプロジェクトデータベースに変換することを検討しているのであれば、 *4D Design Reference* の [データベースをプロジェクトモードに変換する](#) を参照してください。

[変更点とアップデートについて](#)
[廃止予定または削除された機能](#)
[過去のドキュメント](#)

変更点とアップデートについて

R-リリースのサイクルの過程において、4D が使用する様々なライブラリーやコンポーネントはアップデートされ、既存の振る舞いは一部変更となり、既知の問題については記録が残されます。元々は"リリースノート"として発表されていたこれらの情報は、このページに要約されています。

4D アプリケーション

4DRequestsLog フォーマット

- **4DRequestsLog ファイルフォーマット は4D v17R5 以降、変更となりました:**
 - 3つのフィールドが新しく追加になりました。
 - 時間はミリ秒ではなくマイクロ秒単位になりました。

ログ解析ツールを使用している場合、それらをアップデートする必要があるかもしれません。

設定ファイルは自動的に移動されます

設定ファイルのアーキテクチャーの進化を尊重し、4D は一部のデフォルト設定ファイルを自動的に名称変更してその場所を移動させます:

- "BuildApp.xml" は"buildApp.4DSettings" と名称変更され、"Preferences" フォルダからデータベースの"Settings" フォルダへと移動されます。
- "Backup.xml" は"backup.4DSettings" と名称変更され、"Preferences" フォルダからデータベースの"Settings" フォルダへと移動されます。
- 上記の移動の結果Preferences フォルダが空になった場合、それは削除されます。

Current Logs フォルダの位置

- **Logs フォルダの位置が変更になりました:**
 - v18 以降のバージョンにおいて、"カレントの"Logsフォルダはデフォルトでデータファイルと同階層に作成されます。また、メンテナンスログ(圧縮、検証、修復)用の特定のLogs フォルダが、ユーザーのカレントシステムフォルダ(4D フォルダ)内に作成されます。
 - 既存のLogs フォルダはデータベース変換の際、何も操作はされずにそのままにされます。

垂直スケール演算子

インライン・複数行コメント(/*...*/)をサポートするようになった結果、4D v18 以降においてピクチャーの垂直方向スケール演算子に変更になりました。

- 以前の演算子: */
- 新しい演算子: *|

以前の演算子は、データベースが4D v18 以降に変換されたときに自動的に置き換えられます。

範囲チェック

4D v17 R6 以降、4D では範囲チェックは常に有効化されています。その結果、データベース設定ダイアログボックスの"コンパイル"のページから、 **範囲をチェック**チェックボックスが削除されています。

範囲チェックの特殊コメントは引き続きサポートされます:

```
// %R- で範囲チェックを無効化
// %R+ で範囲チェックを有効化
// %R* でデフォルト値にリセット (デフォルト値は設定ダイアログボックスでの設定)
```

// %R* コメントは、今まではデータベース設定でチェックされていなければ"無効化された"状態を再現するのに使用できていたものが、今後は// %R+ と必ず同じ意味になる(オプションが有効化される)という点に注意してください。

フォームリサイズ時にOn Timer は発動しないことに

現在ではサブフォームがリサイズされる際には On Resize イベントがトリガーされるため、 On Timer フォームイベントは衝突

を避けるためにトリガーされないようになりました。以前は、このイベントはトリガーされましたがその頻度が正確性を欠くことがありました。

Form event コマンドの名称変更

以前の **Form event** コマンドは **Form event code** と名称変更され、新たにオブジェクトを返す新コマンド **FORM Event** が実装されました。

Blob 引数の検知の向上

コンパイラーが改善され、BLOB 変数を使用する際のミスマッチをより正確に検知することができるようになりました。コンパイラーは、コマンドがBLOB を要求しているのにBLOB でない引数を受け取った場合に、エラーを表示します。

インデックスのついたオブジェクトフィールドが再インデックスが必要になる可能性

保存されているデータと、それがどう作成されたかによって、オブジェクトフィールドのインデックスが正しくない可能性があります。これは極めて稀な、再現不可能な状況です。一つの要因として見られているのが特殊文字(例: . , "" [] 等) をプロパティ名に使用しているオブジェクトです。これが含まれていると、インデックスが壊れたものとなり、オブジェクトフィールドでの QUERY BY ATTRIBUTE に失敗する現象(ACI0098796) を引き起こす可能性が示唆されています。これらは4D v15 - 4D v17.0 で作成されたデータベースでインデックスのあるオブジェクトフィールドを持つものが対象となっています。v16.4 HF1 / v17.1 / v17R4 以降で作成またはインデックス付がなされたデータベースでは影響ありません。この問題を直すためには、オブジェクトフィールドを再インデックスすることが推奨されています。

新しい4D リモートのリソースキャッシュフォルダ

4D v17 R5 以降、同じアプリケーションの複数のコピー(macOS またはWindows) またはインスタンス(Windows) を実行して同じサーバーへと接続することが可能になりました。これを可能にするため、キャッシュフォルダの命名規則が変わりました。以前のバージョンでの命名規則はサーバーのIPおよびサーバーのポート番号に基づいたものでしたが、これに追加のインスタンス番号が追加されることとなります。古い方のキャッシュフォルダとそのデータは、今後は使用されません。削除したい場合には、恒久的に削除する必要があります。

4D コンパイラーの改善

新しい **Get call chain** コマンドを実装するため、4D コンパイラによって生成されるコードが改善されました。結果として、4D v17 R6 でコンパイルされたデータベースは4D v17 R6 以降のバージョンでしか開けなくなります。新コマンドの利点を引き出すためには、古いバージョンでコンパイルされたデータベースも再コンパイルする必要があります。

Formula コマンドの名称変更

“New formula” および “New formula from string” コマンドは、それぞれ **Formula** および **Formula from string** へと名称変更されました。

On 4D Mobile Authentication データベースメソッドの新しい名前と引数

4D v18 以降、“On 4D Mobile Authentication” データベースメソッドは、以下のように変更になりました：

- **On REST Authentication データベースメソッド** と改名されました。
- 新しい\$4 引数に、呼び出し元のIP アドレスを受け取るようになりました。

4D View Pro

On data Change イベントはトリガーされなくなります

4D View Pro 内の4D イベントはリファクタリングされ、[On Data Change](#) イベントは今後はトリガーされないことになりました。

新しい日付フォーマット

SpreadJS の日付との親和性を高めるため、4D は日付と時間の内部的な保存方法を変更しました。4D v17 R2 以降、4D View を4D View Pro ドキュメントに変換する場合、日付と時間はC_TEXT としてキャストされてきました。4D v17 R4 以降、日付と時間は、次の二つのフィールドを持つC_OBJECT としてキャストされます：一つは日付用のフィールド(C_DATE)、一つは時間用のフィールド(C_TIME) です。

リージョン設定をOSと同じに設定

4D v17 R5 以降、4D View Pro のデフォルトのローカライズ設定は、カレントの4D ホストデータベースと同じ設定になります。

す。そのため4D v17 R5 では、以前のバージョンと比べて4D View Pro のインターフェースでの日付、時間、そして数値の表示が変わっている可能性があります。

4D Write Pro

ページ番号とページ数は常に動的に計算

\$wp_pageNumber および \$wp_pageCount 式は常に計算され、**ST FREEZE EXPRESSIONS** コマンドで固定化することはできません(4D Write プラグインと同じ振る舞いとなりました)。

表のページ分けに関する互換性

以前のバージョンでは、レンジが行またはカラムの内部にあったとき、あるいは改ページが表内に挿入されたとき、改ページが表の前に挿入されてきました。今後は改ページは該当の行の前に挿入され、表は二つに別れることになります。

デフォルトのタブを管理する新しい方法

[wk tab stop offsets](#) の振る舞いが更新されました。タブ引数の最後の要素は、今後はタブオフセットのデフォルト値を定義するためには使用されなくなりました。今後は、最後のタブのオフセット値を定義するようになります。4D v17 R5 以降、デフォルトタブ値は [wk tab default](#) 属性の [wk offset](#) 引数によって管理されるようになります。この属性はデフォルトのタブ値を管理するために作成されました。

OS に関する注記

4D Developer EditionはmacOS High Sierra (10.13) ではブロックされます

macOS High Sierra において4D Developer Edition v18 以降を開こうとした場合、次のような警告メッセージが表示されます:
"4D は開発元が未確認のため開けません。"

macOS Mojave または Catalina を使用することが推奨されます。ただし、macOS High Sierra を開発のために使用しなければならない場合、以下の2つの回避策があります:

- Apple から [Swift](#) をインストールし、4D Developer アプリケーションが正しく認識されるようにします。
- あるいは、"システム環境設定 > セキュリティ"ダイアログボックスを開き、"このまま開く"ボタンをクリックします(これは新しいビルドをダウンロードするたびに実行する必要があるという点に注意してください)。

注: 問題となるのは4D Developer アプリケーションのみです。他のエディション(4D Server、組み込みアプリケーションなど)は特に制約なく直接開くことができます。

Windows 7 とWindows 8 のサポートを終了

Microsoft によるWindows 7 のサポートは2020年の1月で打ち切られます。そのため、4D はそのメンテナンスを4D v17 R5 で打ち切ります。並行して、Windows 8に対するサポートも打ち切りますので、今後はWindows 10 へのアップグレードをご検討ください。

Windows 7 における最小要件

Windows 7: 現在利用可能なMicrosoft アップデートを(システム設定の更新ダイアログを使用して)全てインストールすることが強く推奨されています。KB2999226 のアップデート(2015年9月にリリース)はapi-ms-win-crt-runtime-1-1-0.dllに関連したエラーを回避するためには必須です。Microsoft はすでにWindows 7 のサポートを停止しており、延長サポート(セキュリティ修正)も間もなく終了となります(2020年1月)。Windows 10 へのアップグレードをご検討ください。

macOS Mojave (10.14) での日付と時間式に関するローカライズ問題

macOS Mojave が4D に存在しないローカライズ設定を使用する場合、日付と時間はUS フォーマットで表示される可能性があります。例えば、Mac のクライアントがイタリア語のローカライズを使用していた場合、これに対応する"it.lproj" は存在しませんので、日付はUS フォーマットで表示されることになります。

ライブラリーアップデート

- **openssl:** 4D v18 では **openssl 1.1.1d** にアップデート
- **Xerces:** v3.2.1 にアップデート

- **CEF:** v3626 にアップデート
 - **PHP:** v7.3.1 にアップデート - これはPHP にとってのメジャーアップグレードとなり、追加の機能もありますが互換性の問題を引き起こす可能性もあります。 [詳細な情報についてはこちらをご覧ください](#)。
 - **ICU:** v63.1 にアップデート- この新しいUnicode バージョンへとメジャーアップグレードをしたため、文字、テキスト、オブジェクトのインデックスは強制的に自動リビルドが行われます。
 - **Hunspell:** v1.7.0 にアップデート
-

廃止予定または削除された機能

30年以上にわたり、私たちのゴールは(新しい概念やテクノロジーを用いて)製品を改善していきつつも、4D アプリケーションの互換性を確保することにあります。4D 初期の頃からのユーザーはご存知かもしれませんが、私たちは互換性のためにおおくの労力をさいてきました。そして、古い4D のバージョンやOS を使用して何年も前に作成された4D アプリケーションが、現在も最新バージョンの製品で動作しているケースを、世界中で見つけることができます。

残念ながら、以下のような理由から、旧式のテクノロジーを新しいものと組み合わせるのが困難になる場合もあります：

- 4D では常に新しいテクノロジー、新しいAPI、そして新しい方法論をデベロッパーに届けなければなりません。
- OS は毎日のように変化しています。そして時にはOS が持つ古いAPI が廃止予定となってしまうことがあります。

このため、4D では時として一部のコマンドや機能を廃止予定に分類する必要があります。これはつまり、これらのコマンドや機能は将来のメジャーバージョンにおいて削除される予定であるということを意味します。

何が廃止予定となっていて、代わりにどんな代替機能が利用可能なのかをデベロッパーに知らせることは、その変化をコードに実装するのを手助けするために必要なことです。あらかじめ予告することで、デベロッパーは慌てず焦らず、十分な時間を持って必要な変更を行うことができます。

早見表

"状況"カラムの値についての説明です：

- 削除済み**: カレントのバージョン(または記載されたバージョン)では使用できなくなっています。
- 廃止予定**: 今後使用されるべきではなく、将来のメジャーバージョンにおいて削除される予定です。
- OS**: OSにおいて廃止予定のテクノロジーの状況に抛ります(例:PICT フォーマット)。状況としては"廃止予定"と同等ですが、4Dがそれを削除する前にOSが除去することが有り得ます。

機能	代替手段	カレントの4D のバージョンでの状況
32-bit 版4D/32-bit 版4D Server	64-bit 版4D/64-bit 版4D Server	削除済み
Mac OS QuickDraw フォント	フォント名	削除済み
Web変数に値を自動的に代入する(v13.4以前に作成されたデータベース用の互換性オプション)	WEB GET VARIABLES コマンド(変数を取得)、 WEB GET BODY PART/ WEB Get body part count コマンド(ポストしたファイルを取得)	廃止予定
非Unicodeモード(変換されたv11以前のデータベース)	Unicodeに移行	削除済み
QuickTime サポート(互換性オプション)	ネイティブフォーマットの使用	削除済み
API QuickDraw	サードパーティプラグインには新SDKプラグインを使用して下さい	削除済み
変換されたサブテーブル	N->1 テーブルの使用	廃止予定
XSLT	PHP <i>libxslt</i> モジュールまたは PROCESS 4D TAGS コマンドを使用して下さい	削除済み
Mac リソース	"Resources" フォルダの使用。互換性のために、変換されたデータベース内では使用可能です。ただし書き込みアクセスコマンドはサポートされません	OS (cicnアイコン、 コマンド:削除済み) .RSR ファイルのデータベースは自動的に開かれません。 Open resource file を使用する必要があります。
macOS ファイルおよびフォルダAPI	Path to object 、 Object to path	廃止予定
コードに基づいた標準アクション	文字列の引数(標準アクション テーマ内の"ak"定数)	廃止予定

PICT	新しいフォーマットを使用して下さい。ヘルプに関しては GET PICTURE FORMATS を参照のこと	削除済み
旧式ネットワークレイヤー	ServerNet を使用して下さい	廃止予定
4D Write	4D Write Pro	削除済み
4D View	4D View Pro	削除済み
4D Mobile	ORDA および REST サーバー	廃止予定

ランゲージ: 廃止予定または削除されたコマンド

廃止予定のコマンドは全て"_o_" の接頭辞がつけられ、4D リスト(コードエディター、タイプアヘッド昨日、等)には表示されません。廃止予定コマンドは、既存のコードから削除されることはなく、サポートされる限りは通常通りに動き続けます。廃止予定のコマンドをメソッドに追加することは(推奨されていませんが)可能であり、コマンド名の頭に"_o_" をつけて入力するだけです。これによって正常に解釈されます。

廃止・削除のコマンド	代替機能	廃止時期
4D 環境テーマ: _o_DATA SEGMENT LIST	-	v11
バックアップテーマ: _o_INTEGRATE LOG FILE	INTEGRATE MIRROR LOG FILE	v16
コンパイラーテーマ: _o_ARRAY STRING	ARRAY TEXT	v12
_o_C_GRAPH	(GRAPH コマンドとSVGを併用して下さい)	v12
_o_C_INTEGER	C_LONGINT	v12
_o_C_STRING	C_TEXT (データベースがUnicodeモードになれば使用可能)	v12
データ入力テーマ: _o_ADD SUBRECORD	N->1 リレーションのテーブルにおいて ADD RECORD	v12
Modified	Form event code と <u>On data change</u>	v12
_o_MODIFY SUBRECORD	N->1 リレーションのテーブルにおいて MODIFY RECORD	v12
ドラッグ&ドロップテーマ: _o_DRAG AND DROP PROPERTIES	ペーストボード テーマのコマンド	v17 R4
フォームイベントテーマ: _o_During	Form event code と適切なイベントの組み合わせで置き換え	v12
フォームテーマ: _o_FORM GET PARAMETER	無効化	v17 R4
グラフテーマ: GRAPH (4D Graphエリアを使用)	代わりにSVGピクチャーを使用	v12
階層リストテーマ: _o_REDRAW LIST	コードから除去(v11以降何の動作もしていません)	v11
Language theme: _o_NO TRACE	デバッガのトレースなしボタンを使用してください	v18

リストボックステーマ:		
LISTBOX Get property 定数:		
<u>_o_lk display hor scrollbar/</u>	OBJECT GET SCROLLBAR	v16
<u>_o_lk display ver scrollbar</u>		R3
<u>_o_lk footer height</u>	LISTBOX Get footers height	v16
		R3
<u>_o_lk header height</u>	LISTBOX Get headers height	v16
		R3
<u>_o_lk hor scrollbar position/</u>	OBJECT GET SCROLL POSITION	v16
<u>_o_lk ver scrollbar position</u>		R3
オブジェクト(フォーム)テーマ:		
<u>_o_DISABLE BUTTON/</u>	OBJECT SET ENABLED	v12
<u>_o_ENABLE BUTTON</u>		
<u>_o_OBJECT Get action</u>	OBJECT Get action	v16
		R3
<u>_o_OBJECT SET COLOR</u>	OBJECT SET RGB COLORS	v18
ピクチャーテーマ:		
<u>_o_PICTURE TO GIF</u>	PICTURE TO BLOB	v16
		R5
セクション:		
<u>_o_MOBILE Return selection</u>	ORDA	v18
Printing theme:		
<u>_o_PAGE SETUP</u>	SET PRINT OPTION、 GET PRINT OPTION、 Print settings to BLOB、 BLOB to print settings	v18
SQL テーマ:		
<u>_o_USE EXTERNAL DATABASE</u>	SQL LOGIN	v12
<u>_o_USE INTERNAL DATABASE</u>	SQL LOGOUT	v12
文字列テーマ:		
<u>_o_Mac to Win</u>		v11
<u>_o_Win to Mac</u>		v11
サブレコードテーマ: 全コマンド	“nnn SUBRECORD” と “nnn SUBSELECTION” を、N->1リレーションにあるNテーブルのNレコードまたはNテーブルのアクションで置き換え	v12
システムドキュメントテーマ:		
<u>_o_Document type</u>	Path to object	v12
<u>_o_Document creator</u>	Path to object	v16
		R6
<u>_o_SET DOCUMENT TYPE</u>	Object to path	v16
		R6
<u>_o_SET DOCUMENT CREATOR</u>	Object to path	v16
		R6
<u>_o_MAP FILE TYPES</u>	UTIs および Info.plistを使用	v16
		R6
システム環境テーマ:		
<u>_o_Font name</u>	フォント識別子を使用	v14

OBJECT SET FONT コマンドは、フォントに対し倍長整数の引数を受け取らなくなりました。この引数の型は文字列になり、フォント名を指定する必要があります。

ユーザーフォームテーマ: 全 コマンド		
_o_CREATE USER FORM	ダイナミックフォーム	v17 R4
_o_DELETE USER FORM	ダイナミックフォーム	v17 R4
_o_EDIT FORM	ダイナミックフォーム	v17 R4
_o_LIST USER FORMS	ダイナミックフォーム	v17 R4
4D Internet Commands:		
FTP_Progress	呼び出された場合にはエラーを返す	v16 R2
注: FTP_Append 、 FTP_Receive 、 FTP_Send においては <i>progress</i> 引数はサ ポートされていません		

ランゲージリファレンスの **廃止予定のコマンド** 付録内に、これらのコマンドのアルファベット順の一覧があります。

ランゲージ: 削除されたコマンド

以前は廃止予定であった以下のコマンドは、4D からは削除され今後コードで使用してはいけません。呼び出された場合には、エラー (エラー33 - 未実装のコマンドまたは関数です) がトリガーされます。

警告: 削除されたコマンドへの呼び出しは、シンタックスチェックやコンパイル機能などでは検知されません。ご自身のコードはご自身でチェックし、削除されたコマンドに対して如何なる呼び出しも含まれていないようにしてください。

コマンド	代替機能	削除 時期
4D 環境テーマ:		
_o_ADD DATA SEGMENT	-	v17 R5
グラフテーマ:		
_o_GRAPH TABLE	データを配列でビルドして、SVG ピクチャー内で GRAPH を呼ぶ	v17 R5
ピクチャーテーマ:		
_o_PICTURE TYPE LIST	PICTURE CODEC LIST	v17 R5
_o_QT COMPRESS PICTURE	CONVERT PICTURE	v17 R5
_o_QT COMPRESS PICTURE FILE	WRITE PICTURE FILE/ PICTURE TO BLOB	v17 R5
_o_QT LOAD COMPRESS PICTURE FROM FILE	READ PICTURE FILE/ CONVERT PICTURE	v17 R5
_o_SAVE PICTURE TO FILE	WRITE PICTURE FILE	v17 R5
リソーステーマ:		
_o_ARRAY TO STRING LIST	-	v17 R5
_o_Create resource file	-	v17 R5

_o_DELETE RESOURCE	-	v17 R5
_o_Get component resource ID	-	v17 R5
GET ICON RESOURCE	-	v17 R5
_o_SET PICTURE RESOURCE	-	v17 R5
_o_SET RESOURCE	-	v17 R5
_o_SET RESOURCE NAME	-	v17 R5
_o_SET RESOURCE PROPERTIES	-	v17 R5
_o_SET STRING RESOURCE	-	v17 R5
_o_SET TEXT RESOURCE	-	v17 R5
文字列テーマ:		
_o_Convert case	必要であれば CONVERT FROM TEXT/ Convert to text を使用	v17 R5
_o_ISO to Mac	変換が必要なければコマンドをメソッドから削除してください	v17 R5
_o_Mac to ISO	(これはつまりデータベースがUnicode モードで実行されることを意味します)	v17 R5
システム環境テーマ:		
_o_Font number	フォント識別子を使用	v17 R5
ユーザーインターフェーステーマ:		
_o_Get platform interface/_o_SET PLATFORM INTERFACE	変換されたアプリケーションでのみ、 <u>Automatic Platform</u> 定数を用いることで使用可能	v17 R5
_o_INVERT BACKGROUND	-	v17 R5
Web サーバーテーマ:		
_o_SET CGI EXECUTABLE	-	v17 R5
_o_SET WEB DISPLAY LIMITS	-	v17 R5
_o_SET WEB TIMEOUT	-	v17 R5
_o_Web Context	-	v17 R5
Windows テーマ:		
_o_Open external window	-	v17 R5
XML テーマ:		
_o_XSLT APPLY TRANSFORMATION	PHP <i>libxslt</i> モジュールか、 PROCESS 4D TAGS コマンドを使用してください	v17 R5
_o_XSLT GET ERROR	PHP <i>libxslt</i> モジュールか、 PROCESS 4D TAGS コマンドを使用してください	v17 R5
_o_XSLT SET PARAMETER	PHP <i>libxslt</i> モジュールか、 PROCESS 4D TAGS コマンドを使用してください	v17 R5

旧式のテクノロジー(プロジェクトモードとの整合性)

プロジェクトモードは4D データベースアーキテクチャーにおいて重要な進化の表れです。4D プロジェクトは最新のソフトウェアインターフェース要件と最も現代的なテクノロジーに依存するため、以下にあるような一部の旧式テクノロジーはサポートしません。これらの昨日は現在は廃止予定となっており、将来のプロジェクトアーキテクチャーとより合致するためには今後データベースの中で使用されるべきではありません。

4D のプロジェクトモードアーキテクチャーの詳細な情報については、 [developer.4d.com](https://developer.4d.com/4d_projects_documentation) の [4D projects documentation](https://developer.4d.com/4d_projects_documentation) を参照してください。4D では、4db データベースをプロジェクトモードへと変換する自動書き出しツールも提供しています。書き出しの過程において、旧式の機能は自動的に変換されるか、削除されるか、またはエラーが生成されます([データベースをプロジェクトモードに変換する](#) を参照してください)。

フォームオブジェクトとプロパティ

以下のフォームオブジェクトとプロパティは、カレントのインターフェース要件とは合致しないため、廃止予定となっています。これらは [ダイナミックフォーム](#) ではサポートされず、プロジェクト変換ログファイル内に警告またはエラーを生成します(以下の表のコメントを参照してください)。

廃止予定の機能	変換ステータス	コメント
ハイライトボタン	エラー	3D ボタンに変換する必要あり
ピクチャーラジオボタン	エラー	3D ボタンに変換する必要あり
ダイアル	エラー	進捗インジケーターに変換する必要あり
格子	警告	格子オブジェクトは自動的にsvgピクチャーへと変換され、データベースのresources フォルダに保存されます
ラジオボタンとしてのブールフィールド	警告	サポートされますが、以下の式が割り当てられた標準のグループラジオボタンの組へと自動的に変換されます: <code>[table]Boolean_field</code> および <code>Not([table]Boolean_field)</code>
On Background ピクチャーフォーマット	-	"トランケート(中央合わせしない)"へと変換されます
リストボックス - スクロール可能エリアとの互換性	警告/エラー	標準のリストボックス機能を使用してください
リストボックス - 接続されたリストボックスとの互換性	エラー	標準のリストボックス機能を使用してください
プラットフォームインターフェースの"printing" プロパティ	警告	"printing" プロパティを持つオブジェクトは自動的に"flat" スタイルへと変更されます ("system" 境界線を持つボタン、チェックボックス、ラジオボタン、変数/フィールド)

データベースストラクチャーオプション

以下のデータベースストラクチャーオプションは廃止予定となっており、編集されるか、プロジェクト変換ログファイル内にエラーを生成します(以下の表のコメントを参照してください)。

廃止予定の機能	変換ステータス	コメント
"修正不可"フィールドオプション	警告	プロジェクトへの書き出しの過程でフォームレベルへと自動的に移動
"表示のみ"フィールドオプション	警告	プロジェクトへの書き出しの過程でフォームレベルへと自動的に移動
"必須入力" フィールドオプション	エラー	"NULL 値の入力を拒否"オプションを選択してください

ツールボックス

以下のツールボックスエディターや機能は、廃止予定となっており、プロジェクトではサポートされません:

廃止予定の機能	変換ステータス	コメント
ピクチャーライブラリー	警告	ピクチャーは自動的にデータベースのResources フォルダへと書き出されます
GET PICTURE FROM LIBRARY	-	動作しません。代わりに READ PICTURE FILE を使用してください。
"ユーザー更新可" リストオプション	-	
LIST OF CHOICE LISTS	-	-

SAVE LIST

-	-	プロジェクトから呼び出された場合にはランタイムにエラー
"" または "*" と名前がついたグループ	エラー	プロジェクトモードではこれらは予約されています
"Designer" または "Administrator" という標準ユーザー名	-	これらの名前はプロジェクトモードではこれらは予約されています

互換性の設定

旧式の互換性モードはプロジェクトではサポートされません。互換性の設定はプロジェクトへの書き出しの過程で新規データベースと同様にリセットされます。お使いのデータベースが互換性設定に依存している場合、アップデートを検討してください。

4D では [互換性の設定をどのように取り除いたらいいのかという点に特化したブログ記事の一連](#) を読むことを強く推奨します。

4D v18の時点で削除済みまたは廃止予定のもの

32-bit版

4D v17 R5 以降、32-bit 版の4D および4D Server は製作も配信もされません。 今後は64-bit 版のみが利用可能となり、これによって4D アプリケーションはこの強力なアーキテクチャーの利点を全て活かしきることができるようになります。

64-bit 版では旧式の4D の機能をほぼサポートしていますが、64-bit 版では最新のテクノロジーを採用しており、以前のバージョンで廃止予定と宣言されていたものに関してはサポートしません。32-bit 版で開発されたデータベースをアップグレードする場合にはいくつかの調整が必要になる場合があります。データベースを調整するには、v17 への変換マニュアルの"32-bit 版から64-bit 版への変換について"を参照して下さい。

XSLT コマンド

XSLTは4D では使用できなくなりました。XSLT コマンドを呼び出した場合、エラー33 "未実装のコマンドまたは関数です"が返されます。

しかし4D 引き続きXSLT をお使いのお客様のために、PHP XSL ライブラリーに依存する決定をしました。これはXSL 変換に必要なオペレーションを全て実行できる包括的なAPI を提供しています。このライブラリーは効率的なツールで、**_o_XSLT APPLY TRANSFORMATION**、**_o_XSLT SET PARAMETER** および **_o_XSLT GET ERROR** が削除されたあと、簡単に置き換えることが可能です。4Dでは、4D XSLT コマンドの代わりにPHP XSLを代用するための詳細をまとめたドキュメントをご用意しています: [Download XSLT with PHP technical document](#) (PDF).

また、HTMLページの動的な生成を扱う際には4Dタグの使用を検討することが推奨されます。何故なら、HTMLコードをフォーマットされていないテキスト として扱っている場合には、多くの場合こちらの方が簡単だからです(**PROCESS 4D TAGS** コマンドも参照して下さい)。

QuickTime

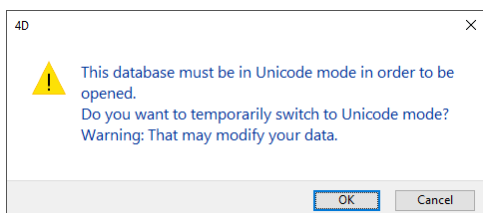
(database parameter を通しての)QuickTime のサポートは、4D v17 R5 以降では削除されます。QuickTime に関連したピクチャーコーデックは今後macOS においてもWindows においてもサポートされなくなります。

フロート型フィールド

4D v18 以降、フロート型フィールドは4D データベースではサポートされません。この型のフィールドは4D SQL エンジンでしか使用されていませんでした。データベースの中でフロート型フィールドを使用している場合、これらはデータベースがv18 以降のバージョンで開かれる際に自動的に実数型へと変換されます。既存のデータについては、圧縮をかける、またはレコードを再保存しない限りは変換はされません。

非Unicodeモード

ASCII モード(この場合"非Unicodeモード"と同義)のサポートは4D v17 R5 以降では削除されます。Unicode モードに設定されていない変換されたデータベースを開いた場合、4D は以下のようなダイアログボックスを表示します:



このアラート画面は、データベース設定の"互換性"ページの **Unicodeモード**がチェックされていない限り、毎回開始時に表示されます。変換されたデータベースにおいては、必ずUnicode モードを有効化する必要があります。 [Conversion to 4D v14](#) のPDF ドキュメントがこの件についてのヒントを提供しています。

PICT フォーマットのピクチャー

PICT フォーマット(マクロのエンコーディング)

PICT フォーマット(Appleが2005年に廃止予定としたフォーマット)は、v17 R5 以降、4D ではサポートされなくなります。これはPICT フォーマットのピクチャーはどのような手段でもレンダリングされないため、変換される必要があるということです。

GET PICTURE FORMATS コマンドを使用して、データファイル内からPICT フォーマットを検知してフィルタリングしましょう。

DatabaseOutsidePackage キー

info.plist 内の *DatabaseOutsidePackage* キーを使用した、mac プラットフォーム上のシングルユーザー向け組み込み4D アプリケーションにでの特殊な配布モードは、今後サポートされません。4D v17 R5 以降、*DatabaseOutsidePackage* キーは無視されます。

Right-to-left 機能の削除

4D v17 R4 以降、4D ではWindows のRight-to-left インターフェースに対する特定のサポートを提供しないことになりました。対応するオプションは、データベース設定からもフォームプロパティからも利用できなくなります。変換されたデータベースにおいては、Right-to-left 設定はランタイムでは無視されます。

ユーザーフォーム

4D v17 R4 以降、ユーザーフォームは廃止予定となります。カスタマイズ可能なユーザーインターフェースを提供するためには、今後は **ダイナミックフォーム**を使用することが推奨されます(**デザインリファレンス**の **ダイナミックフォーム**を参照してください)。"ユーザーフォーム"テーマ内のコマンドは全て廃止予定に指定されました。

ユーザーフォームは互換性のために引き続きサポートはされますが、今後は使用されるべきではありません。ユーザーフォームのサポートは将来のリリースで打ち切られる予定です。

4D Mobile

4D Mobile 機能は4D v18 以降廃止予定であり、今後は使用されるべきではありません。REST リクエスト経由での4D データへのアクセスは、今後はORDA テクノロジーと4D REST サーバーによってサポートされます。

以前のリリースで既に廃止予定だったもの

Web変数に値を自動的に代入する

以前のバージョンの4Dでは、WebフォームまたはURLを通して送られた変数の値が、4D内の変数と同じ名前であったときに、Webサーバーは自動的にその値を4D変数へと代入していました。

最適化と管理の目的から、この原理は4D v14以降は維持されていません。Web変数の値は自動的に4D変数に代入されません。POST や GET を使用して送られた変数を取り戻すためには、WEB GET VARIABLES コマンドを必ず使用して下さい。送られたファイルを取り戻すためには、WEB GET BODY PART/WEB Get body part count コマンドを使用しなければなりません。

注: 自動的な代入は、v13.4以降で作成された4Dデータベースにおいてもデフォルトで無効化されています。

しかしながら、互換性のために、この機構はv13.4より前のバージョンの4Dで作成されたデータベースにおいてはデフォルトで維持されています。この場合、データベース設定の互換性ページの「Web変数に値を自動的に代入する」のオプションを使用して無効化する事ができます。

この機構は廃止されていることから、今後の変換を容易にするためにも、変換されたデータベースにおいてこのオプションのチェックを外すこと(またそれに加えて必要であればコードを修正すること)が強く推奨されます。

サブテーブル

これまでの複数のメジャーバージョンにおいて、4Dはデベロッパに対し、サブテーブルの使用をやめるよう忠告をしてきました。そして4D v11以降、サブテーブル型のフィールドを作成することは出来なくなりました。サブレコードにはいくつかの制約が付きまします。例えば、サブテーブルは必ずメモリーにロードされ、**SEND RECORD** と **DUPLICATE RECORD** コマンドで管理することはできません。

今すぐサブテーブルのサポートを打ち切る予定はありませんが、デベロッパの方々にとって、そろそろ真剣にサブテーブルをN対1テーブルへと切り替えるべき時期だと考えられます。何故なら、将来の4Dのメジャーバージョンにおいて4Dからサブテーブルが除去される予定だからです。パフォーマンスのため(リレートしたレコードの読み出しが遅いなどの特定の条件下)にサブテーブルを使用していたデベロッパは、ご安心ください。スピードは(特にv12以降)N対1、1対Nリレーションを使った方が速いです。サブテーブルを除去するためには、主に二つのステップがあります(注:以下は完全なTech Tipsではなく、あくまで早見表です):

- v11以前のストラクチャーを変換する前に: 2004では、適当なNテーブルを作成し、1テーブルにIDフィールドを作成します(作成されていなかった場合)。その後変える必要のあるコードを全て書き換えます(以下を参照して下さい)。

- 変換後: ここでは、4Dはサブテーブルを特殊なリレーションを使用してNテーブルに置き換え、ランゲージがサブセレクションとサブレコードに作用するようになって います。4Dデベロッパはこの特殊なリレーションを削除し、通常のリレーションと置き換えた上で必要なコードを全て書き換えます(以下を参照して下さい)。

ここで言う"必要なコードを全て書き換える"というのは、基本的に:

- 新しいフォームを作成し、含まれるフォームをアップデート
- メソッド(プロジェクトメソッド、フォームメソッド、オブジェクトメソッド、等)において:
 - "サブレコード"テーマのコマンドを全て除去し、対応するセレクションまたはレコードコマンドで置き換えます(例えば、 **_o_CREATE SUBRECORD** を **CREATE RECORD** コマンドで置き換え、IDフィールドを埋めます)
 - 必要に応じてNレコードを明示的に読み出します。

注: 4D v14 R3以降、サブテーブルを含むデータベースを変換した際に4Dによって自動的に追加される特殊なフィールド" id_added_by_converter"に値を割り当てることが出来るようになります。これにより、"サブテーブルリレーション"のリンクを保ったまま、 **_o_CREATE SUBRECORD** などの廃止予定のコマンドを使用することなく、リレートされたレコードを追加または編集することができます。メソッドをアップデートした後、これらの特殊なリレーションは、コードを何も書き換えることなく、通常のリレーションと置き換えることができるようになります。

フォント番号

QuickDraw フォント(例: Geneva、Chicagoなど)は廃止予定となり、今後はフォントを指定するのにID番号を使用するべきではありません。 **_o_Font name** コマンドは互換性のために残されていますが、今後のバージョンでは削除されます。今後は **OBJECT SET FONT** コマンドはフォント名のみを受け付けます。

プラグインのためのAPI QuickDraw

プラグインには二つの種類があります: 新しいプラグインAPIを使用するものと、古いAPI(とQuickDraw)を使用するものです。古いツールボックス(とQuickDraw)を使用するプラグインに対しては、互換性を維持するため、描画/レンダリングは以前のバージョンのように QuickDraw ポート内で直接行われることはありません。その代り、プラグインに割り当て有られたGWorld QuickDraw オフスクリーンエリアを通して行われます。

その結果、プラグインはコンテナ(フォームオブジェクト)によって設定されたカレントのポートを変更してはいけな、などのいくつかのルールに従う必要があります。

新しいツールボックスを使用するプラグインに関しては、その新しいツールボックスのみが使用され、QuickDraw は使用されません (<https://github.com/4D/4D-Plugin-SDK> を参照して下さい)。

macOS でのファイルとフォルダAPI

4D v16 R6以降、Mac OS 9ベースのファイル・フォルダ管理のためのAPIは、4Dにおいては廃止予定となります。これらのAPIは既にAppleによってかなり以前から廃止予定とされていたものでした。

古いAPIに依存していた以下の4Dコマンドは、名前が変更されています: **_o_Document type**、**_o_Document creator**、**_o_SET DOCUMENT TYPE**、**_o_SET DOCUMENT CREATOR**、および **_o_MAP FILE TYPES**。macOS用32-bit版4D においては、これらのコマンドは互換性のために維持されていますが、廃止予定となっており、今後使用するべきではありません。4Dではパス名とファイル拡張子を管理するための、二つの新しいコマンドが追加されました。 **Path to object** および **Object to path** です。

コードに基づいた標準アクション

標準アクションは4D v16 R3において内部的に設計し直されました。今後は文字列パターンに基づき、また引数をサポートする様になります(詳細な情報については **標準アクション**の章を参照してください)

変換されたアプリケーションにおいては、標準アクションは透過的に新デザインへとリマップされます。しかしながら、4Dランゲージにおいては、標準アクションは以前はコードの値に基づいていました。この原理は廃止予定となり、今後は使用されるべきではありません:

- **標準アクション** 定数テーマ内にあるコードベースのテキスト標準アクション(以前の名前は"関連付けられた標準アクションのテキスト値")は全て接頭辞"**_o_**"がつき、今後は使用されるべきではありません。
- **割り当てられた標準アクションの値** 定数テーマは全て廃止予定となります。
- (コードを返す) **_o_OBJECT Get action** コマンドは廃止予定となり、新しくなった **OBJECT Get action** コマンドによって置き換えられます(こちらは名前を返します)。

Macリソース

これもまた古いMac OSのテクノロジーの一つで、Mac OS X 10.4(Tiger、2005年)に廃止されています。リソースは現在はテキストや文字列などの構造化されたデータ(ローカライズ用)に加え、アイコンなどを保存するために使用されています。つまり廃止

されたのはリソースそのものではなくて、リソースフォークと呼ばれる、ディスク上のサポートです。リソースフォークはMac OS ファイルシステムの一部であり、Mac OS X初期から、Apple ではこのサポートを除去しようと試みてきました。これはこの仕組みがUnixやWindowsなどの他のファイルシステムとは互換性がないうえに、ファイルをネットワーク越しに転送する際に多くの問題の原因となってきたからです。

Windows上では、このメカニズムをエミュレートして、Macリソースは.RSR ファイル内に置かれてきました。

しかし、これらのリソースを管理するAPIがまだあるとはいえ(そしてMac OSではデータフォーク内に保存されているリソースをひそかに管理しているとはいえ)、以下の複数の理由からこの機構は使用されるべきではありません:

- テキストと文字列はMac-Romanになっています。TEXT または STR# タイプのリソースにはUnicodeを保存することができません。
- PICT リソースは PICT フォーマットを保存します。これは現代的ではなく、透明度もありません(上記の"PICT フォーマットのピクチャー"を参照して下さい)。
- リソースには数・サイズともに上限があります(約2,700リソース、または16MB)。

4Dからはリソースを書き込み・作成するコマンドは除去されています。

リソースを依然として使用している4Dアプリケーションの大半は、実際に使用しているのは"String List"リソース('STR#')です。4Dでは STR# を XLIFF に容易に変換するツールを提供しています:

- 4D Pop コンポーネントは、STR# ファイルの中身を読んで変換することで自動的に XLIFF ファイルを作成する事ができます。
- STR# を参照する全てのルーチンと式は、何の変換もなしにそのまま XLIFF を使用する事ができます。例えば、ボタンやメニューのラベルが":15000,3" (=「STR# ID 15000 の3つめの項目を取得しなさい」という意味) だった場合、4D は適切な XLIFF を(存在すれば)読み込みます。

他のリソースに関して:

- リソースをResources フォルダ内の他のファイル形式に格納します(必要であればサブディレクトリを作成して下さい):
 - 'TEXT' リソースを XLIFF または .txt ファイル形式に保存します。
 - 'PICT' リソースを別の.jpg/.png などのファイル形式に保存します。
 - 'PICT' + MASK' リソースを .png ファイルとして保存します。
 - 他のプライベートリソースを、適切な形で保存します(多くの場合、特定の拡張子付きのバイナリーファイルとして保存します)。
- "Resources" フォルダにリソースを保管します。リソースの親のパスを動的に取得する場合には **Get 4D folder(Current resources folder)** コマンドを使用します。

CPU優先度の管理

CPU優先度の変更は廃止予定の設定となりました。4Dアプリケーションのメモリ管理は異なるバージョンを経るごとに最適化されてきており、デフォルトの設定を変更することは不要であるばかりか時として非生産的な結果になることもあります。結果として:

- (データベース設定の"一般"ページ内にある)"CPU優先度"エリアは、変換されたデータベース内においてカスタムの値が事前に設定されていた場合に限り表示されます。またこの場合でも **初期設定** ボタンを押してこの値を初期化し、ダイアログボックスから削除することが推奨されます。
 - 4D v16 R2以降、**SET DATABASE PARAMETER** コマンドにおける廃止予定のセクター(_o_4D Remote mode scheduler、 _o_4D Local mode scheduler および _o_4D Server scheduler)は改名され、今後は使用されるべきではありません。
-

過去のドキュメント

このドキュメントは、 **4D v18 プロダクトレンジ**に関するドキュメントです。参照のために、以前のプロダクトレンジで廃止になった機能について言及している過去のドキュメント(PDF、英文)を閲覧することができます。リンクは以下の通りです:

- [Deprecated and Removed Features in 4D v17 - \(Rev. Feb 2018\)](#)
 - [Deprecated and Removed Features in 4D v16 - \(Rev. Jan 2017\)](#)
 - [Deprecated and Removed Features in 4D v15 - \(Rev. June 2015\)](#)
 - [Deprecated and Removed Features in 4D v14 - \(Rev. Oct 2014\)](#)
 - [Deprecated and Removed Features in 4D v13 - \(Rev. 20 Feb 2012\)](#)
 - [Deprecated and Removed Features in 4D v12 - \(Rev. 03 Jun 2010\)](#)
-